IBM Enterprise Storage Server™

SCSI Command Reference
2105 Models E10, E20, F10, and F20

IBM Enterprise Storage Server™

SCSI Command Reference
2105 Models E10, E20, F10, and F20

> **Note!**
>
> Before using this information and the product it supports, be sure to read the "Safety and environmental notices" on page xi and the general information under "Notices" on page 131.

Second Edition (June 2000)

This edition replaces SC26-7297-00.

Publications are not stocked at the address given below. If you want additional IBM publications, ask your IBM representative or write to the IBM branch office serving your locality.

A form for your comments is provided at the back of this document. If the form has been removed, you may address comments to:

International Business Machines Corporation
RCF Processing Department
G26/050
5600 Cottle Road
San Jose, CA 95193-0001
U.S.A.

FAX: 1-800-426-6209

You can also send your comments electronically to:

starpubs@us.ibm.com

For additional information on IBM storage products, visit the Web site at:

http://www.ibm.com/storage

Contents

Figures

Safety and environmental notices

This section contains information about:

- Safety notices used in the ESS publications
- Environmental guidelines for this product

Safety notices

For a translation of the danger, caution, and attention notices, see *IBM Storage Solution Safety Notices*, GC26-7229.

Translated notices are easy to locate in the safety notice translations as they are in numeric order. Look for the ID numbers (**1000** and **1001**) in the following examples.

DANGER

> **A danger notice indicates the presence of a hazard that has the potential of causing death or serious personal injury.**
>
> **1000**

CAUTION:
A caution notice indicates the presence of a hazard that has the potential of causing moderate or minor personal injury.

1001

Attention:
An attention notice indicates the possibility of damage to a program, device, system, or data.

1002

Product recycling

This unit contains recyclable materials. These materials should be recycled where processing sites are available and according to local regulations. In some areas IBM provides a product take-back program that ensures proper handling of the product. Contact your IBM representative for more information.

Product disposal

This unit may contain batteries. Remove and discard these batteries, or recycle them, according to local regulations.

About this book

This publication describes the functions of the IBM Enterprise Storage Server (ESS) and gives reference information such as channel commands, sense bytes, and error recovery procedures.

A glossary at the back of this book defines the terms used in this book. This glossary includes terms for other Seascape solution products.

Who should use this book

This publication documents command parameters and status information particular to the ESS. This information is useful to programmers and systems programmers who are writing programs that issue SCSI commands to the ESS. This information is also useful to individuals who are tracing the flow of information on a ESS SCSI bus.

You can install and operate the ESS in supported environments without knowledge of the information in this publication.

Publications

This section describes the ESS library and publications for related products. It also gives ordering information for these publications.

The ESS product library

The ESS is an IBM Seascape™ architecture-based product. See the following publications for more information on the ESS:

- *IBM Enterprise Storage Server Introduction and Planning Guide, Models E10 and E20*, GC26-7294

  This guide introduces the product and lists the features you can order. It also provides guidelines on planning for the installation and configuration of the ESS.

- *IBM Enterprise Storage Server User's Guide, Models E10 and E20*, SC26-7295

  This guide provides instructions for setting up and operating the ESS.

- *IBM Enterprise Storage Server Host Systems Attachment Guide, Models E10 and E20*, SC26-7296

  This guide provides guidelines for attaching the ESS to your host system.

- *IBM Enterprise Storage Server SCSI Command Reference, Models E10 and E20*, SC26-7297

  This book describes the functions of the ESS. It provides reference information for UNIX™ and Application System/400® (AS/400®) hosts, such as channel commands, sense bytes, and error recovery procedures.

- *IBM Enterprise Storage Server System/390 Command Reference, Models E10 and E20*, SC26-7298

This book describes the functions of the ESS and provides reference information for System/390® hosts, such as channel commands, sense bytes, and error recovery procedures.

- *IBM Enterprise Storage Server Web Interface User's Guide*, SC26-7346

  This book provides instructions for using the IBM StorWatch™ Enterprise Storage Server Web interface, ESS Specialist.

- *IBM Storage Solutions Safety Notices*, GC26-7229

  This book provides translations of the danger notices and caution notices that IBM uses in ESS publications.

- *IBM Enterprise Storage Server Configuration Planner*, SC26-7353

  This book provides work sheets for planning the logical configuration of the ESS. This book is not available in hard copy. This book is only available on the following Web site:

  http://www.ibm.com/storage/ess/ess.htm

- *IBM Enterprise Storage Server Quick Configuration Guide*, SC26-7354

  This booklet provides flow charts for using the StorWatch Enterprise Storage Server Specialist. The flow charts provide a high-level view of the tasks the IBM service support representative performs during initial logical configuration. You can also use the flow charts for tasks that you might perform during modification of the logical configuration.

  The hard copy print of this booklet is a 9-inch X 4-inch fanfold.

Ordering publications

All of the publications that are listed in the "The ESS product library" on page xiii are available on a compact disc that comes with the ESS, unless otherwise noted. You can also order a hard copy of each publication. For publications on compact disc, order the *IBM Enterprise Storage Server Customer Documents compact disc*, SK2T-8770.

All the customer documents listed above are available on the following ESS Web site:

http://www.ibm.com/storage/ess/ess.htm

Related publications

The following related publications are also available:

- *DFSMS/MVS V1R5 Software Support for the IBM Enterprise Storage Server*, SC26-7318

  This book describes the changes you need to make to operating systems for System/390 hosts for support of the ESS functions.

- *IBM Subsystem Device Driver*,

  This book describes how to use the IBM Subsystem Device Driver on open host systems to enhance performance and availability on the ESS. The IBM subsystem device driver creates redundant paths for shared LUNs. The subsystem device driver permits applications to run without interruption when path errors occur. It balances the work load across paths. And, it integrates with applications transparently.

This publication is not available in hard copy. However, it is included on the compact disc that is delivered with your ESS. You can also obtain a copy from the following Web site:

http:www.storage.ibm.com/hardsoft/products/ess/ess.htm

- *IBM Storage Area Network Data Gateway Installation and User's Guide, 2108 Model G07*, SC26-7304

  This book introduces the IBM Storage Area Network (SAN) Data Gateway and provides the procedures for installing and using it.

- *IBM Advanced Copy Services*, SC35-0355

  This book provides detailed information on Copy Services functions such as peer-to-peer remote copy (PPRC), extended remote copy (XRC), and FlashCopy.

- *IBM Enterprise Storage Server*, SC24-5465

  This book, from the IBM International Technical Support Organization, describes the ESS, its components, and its functions.

- *The IBM Enterprise Storage Server in Your Environment*, SG24-5420

  This book, from the IBM International Technical Support Organization, provides information about ESS performance in various environments.

- *IBM S/360, S/370, and S/390® Channel to Controller Original Equipment Manufacture's Information*, GA22-6974

  This publication provides a description of the interface between S/360, S/370™, and S/390® channels and controllers that are designed to operate with this interface. This publication covers the parallel I/O interface. It does not cover the interface between the controller and the I/O device, nor does it cover the ESCON-I/O interface for the S/390.

  This publication is for designers of programs and equipment that are associated with parallel I/O interfaces, and for service personnel that maintain the equipment. Anyone concerned with the functional details of this interface will also find it useful.

- *Fibre Transport Services (FTS) Direct Attach, Physical and Configuration Planning*, GA22-7234

  This publication provides information on fibre optic and ESCON® trunking systems.

Web sites

Additional information is available on the following Web sites:

- IBM Storage:

  Visit the SUPPORT pages at the IBM Storage home page:

  http://www.storage.ibm.com/

  View and print IBM Enterprise Storage Server publications, and get specific details on models and operating systems that the ESS supports:

  http://www.ibm.com/storage/ess

Summary of Changes

Vertical revision bars in the left margin indicate technical changes to this document. Minor editorial changes are not marked with vertical revision bars.

June 2000

The changes for this publication include the following:

- Changes have been made to Chapter 1, "IBM Enterprise Storage Server overview" on page 1 to include overview information on the Models F10 and F20.

- A new chapter has been added to include information on fibre channel. Refer to Chapter 9, "Fibre channel" on page 89.

Chapter 1. IBM Enterprise Storage Server overview

This section provides an overview of the IBM Enterprise Storage Server (ESS) and describes its interfaces and components.

The ESS is a member of the Seascape™ product family of storage servers. The ESS provides integrated caching and a choice of either 8 GB or 16 GB cache. It also provides redundant array of independent disks (RAID) support for the disk drive modules (DDMs). The DDMS are attached through a serial storage architecture (SSA) interface.

The ESS provides:

- RAID 5 or non-RAID

- Fast SSA drives

- Fast RISC processors

- Fault tolerant system

- Storage sharing S/390 and open systems

- OS/390® parallel I/O

- Instant copy solutions

- Disaster recovery solutions

The Model E20 with expansion enclosures provides up to 11 terabytes (TB)[1] of storage capacity, with a choice of 9.1, 18.2, or 36.4 gigabyte (GB)[2] DDMs. See Figure 1 on page 2 and Figure 2 on page 2 for illustrations of the ESS models and the expansion enclosure. When referring to DDM capacity, terabytes (TB) means 1 000 000 000 000 bytes; total user-accessible capacity may vary depending on operating environment. When referring to DDM capacity, gigabyte (GB) means 1 000 000 000 bytes; total user-accessible capacity may vary depending on the operating environment.

The ESS Model E10 does not support an expansion enclosure. ESS Models F10 and F20 have a faster processor than the ESS Models E10 and E20. They also offer more cache, 8 GB or 16 GB.

---

[1] For processor storage, real and virtual storage, and channel volume, a terabyte equals 2(40) or 1 099 511 627 766 bytes. For disk storage capacity and communications volume, a terabyte equals 1 000 000 000 000 bytes.

[2] A gigabyte of storage is 10⁹ bytes. A gigabyte of memory is 2(30) bytes.

Front View　　　　　　　　　Rear View
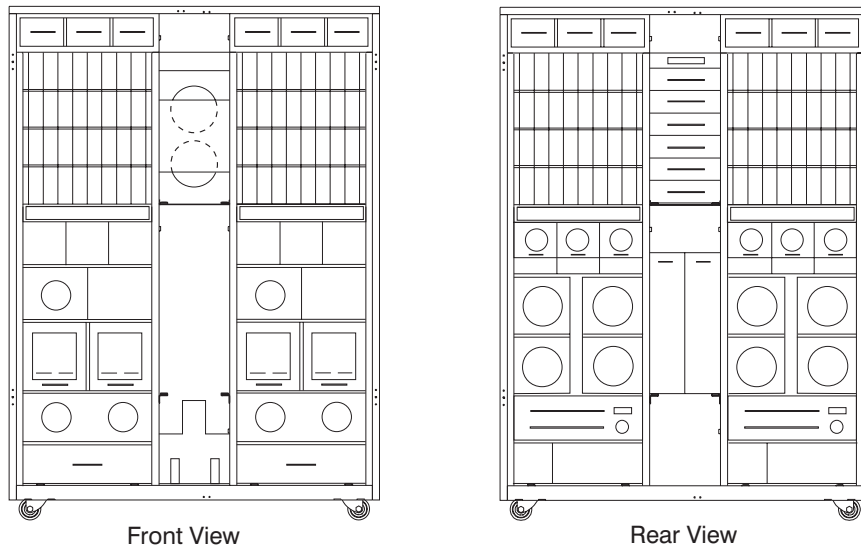
Figure 1. Models E10, E20, F10 and F20 front and rear views

Each ESS expansion enclosure has dual power cables and redundant power. The redundant power system allows both the storage controller and DDMs to continue normal operation when one of the power cables is inactive. Redundancy also ensures continuous data availability.

Front View　　　　　　　　　Rear View
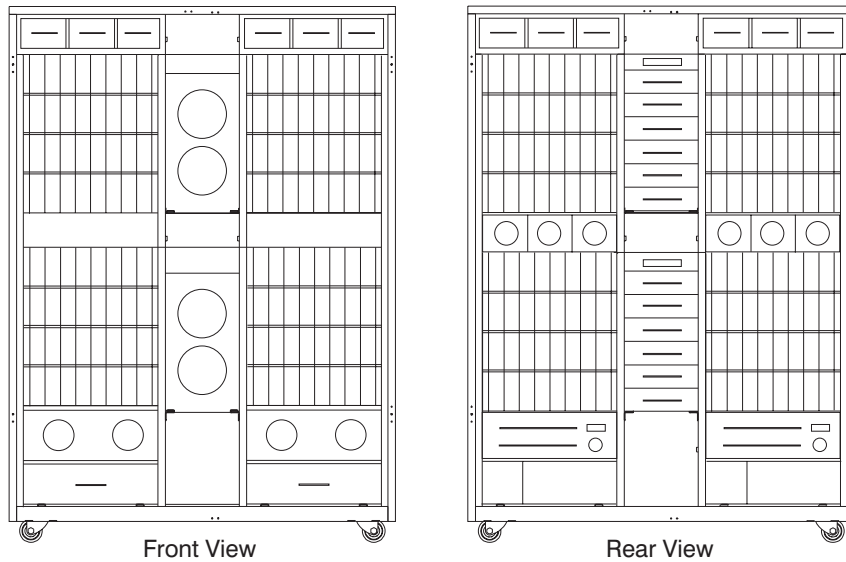
Figure 2. Expansion enclosure, front and rear views

The ESS Model E10 supports a maximum of 64 DDMs. The ESS Model E20 supports a maximum of 128 DDMs. The Model E20, with the expansion enclosure, supports a maximum of 384 DDMs.

Host systems supported by the IBM ESS

The following sections contain information for attaching your ESS to:

- Open systems (SCSI attachment)
- Short wave fibre-channel attachment to an open host system
- S/390

SCSI host systems

The ESS provides heterogeneous data storage that you can share with open host system (SCSI and fibre-channel attachment[3]) and System/390 work loads. The ESS supports the following interfaces and host systems. With SCSI adapters you can connect up to 128 host systems (four per SCSI interface).

The following host systems support SCSI attachment:

Note: See the Web site http://www.ibm.com/storage/ess/ess.htm for details about the types, models, adapters and operating systems supported for SCSI host systems.

- Hewlett Packard (HP-UX operating system)
- IBM RISC System/6000™ and RISC System/6000 SP™ (IBM AIX™ operating system)
- IBM AS/400 (IBM Operating System/400 Version 3 or Version 4 (OS/400®) operating system)
- IBM Netfinity® and Intel-based PC servers (Microsoft™ Windows NT™ operating systems)
- Sun (Solaris operating system)
- Data General (DG UX operating system)
- Intel-based PC servers (Novell Netware)
- Compaq AlphaServers (TRU64 UNIX and OpenVMS)

Fibre-channel host systems

Each fibre-channel adapter has one port and can support one host.

The following host systems support short wave fibre-channel attachment to an open host system:

Note: See the Web site http://www.ibm.com/storage/ess/ess.htm for details about the types, models, adapters and operating systems supported for fibre-channel host systems.

- Hewlett Packard (HP-UX operating system)
- IBM RISC System/6000 and RISC System/6000 SP (IBM AIX operating system
- IBM Netfinity and Intel-based PC servers (Microsoft Windows NT operating systems)
- Sun (Solaris operating system)

---

[3] The ESS will support fibre-channel attachment for open host systems in the second quarter of 2000.

- Data General (DG UX operating system)
- Intel-based PC servers (Novell Netware)
- Compaq AlphaServers (TRU64 UNIX and OpenVMS)

S/390 host systems

With ESCON adapters, you can have up to 32 connections, each with up to 64 logical paths.

The following S/390 ESCON models support the following operating systems:

Note: See the Web site http://www.ibm.com/storage/ess for details about the functional modes and software levels supported for S/390.

- MVS™
- VM
- VSE
- TPF

Web interfaces

This section describes the Web security, the ESSNet, and the Web interfaces for ESS. The Web interfaces include:

- StorWatch Enterprise Storage Server Specialist (ESS Specialist)
- StorWatch Enterprise Storage Server Specialist Copy Services (ESS Copy Services), an optional ESS feature
- StorWatch Enterprise Storage Server Specialist Expert (ESS Expert), an optional software product

See your IBM field office for information about how to order the optional Web interfaces. See the *IBM Enterprise Storage Server Web Interface User's Guide* for detailed descriptions of the ESS Specialist and the ESS Copy Services Web interface and instructions about how to use the interfaces.

Web connection security

You connect to the ESS through the ESSNet.

All data that is sent between the ESS and the Web browser through the ESSNet is encrypted to avoid unauthorized modification of configuration commands. Passwords and authorization levels protect access to the interface.

You control user access by assigning levels of access and passwords.

IBM Enterprise Storage Server Network

The IBM Enterprise Storage Server Network (ESSNet) is a private network resident on an IBM workstation. It is a required feature. IBM installs the ESSNet when they install your IBM Enterprise Storage Server. ESSNet hardware includes:

- The IBM workstation (a PC) and monitor
- An external Ethernet hub that provides cable connections from the ESSNet to the IBM Enterprise Storage Server

Note: You can attach your Ethernet LAN to the external hub. You provide any hardware that is needed for this connection.

- A modem and a modem expander that allow your ESS to communicate between you and IBM

   Note: Remote service support includes this equipment.

The ESSNet software on the workstation includes:

- Windows NT 4.0 operating system

- Browser software (Microsoft Internet Explorer) that allows access to the ESS Specialist

- The ESSNet application for installation and configuration. The ESSNet workstation includes an application that provides links to the ESS. Clicking one of these links initiates the ESS Specialist

The ESSNet provides:

- Support for multiple ESSs. A hub with 16 ports will support 7 ESSs.

- Connectivity between your ESS and the ESS Specialist Web interface. The ESSNet provides browser software at the appropriate level for this connection.

- Improvements in Web performance compared to other Web browser connections.

- Faster network connectivity and elimination of network setup problems when using other Web connections.

- Ethernet connectivity through an Ethernet hub to the ESSNet.

- An independent platform that facilitates installation and configuration of the ESS.

- Software for maintenance and configuration.

IBM installs the ESSNet when they install your first IBM Enterprise Storage Server.

Accessing ESS Specialist and ESS Copy Services

You access the IBM StorWatch Enterprise Storage Server Specialist (ESS Specialist) and IBM StorWatch Enterprise Storage Server Copy Services (ESS Copy Services) from the ESSNet. The ESSNet includes browser software for this access.

You access ESS Copy Services from ESS Specialist.

You can access ESS Specialist from your own workstation if you connect the ESSNet into your intranet using one of the ports on the ESSNet hub. Access ESS Specialist using a browser that is fully enabled for Java™ 1.1, such as Netscape Navigator 4.5 or Microsoft Internet Explorer 4.

If you do not choose to attach ESSNet into your intranet, then you can only access ESS Specialist from Microsoft Internet Explorer on the ESSNet workstation (an IBM personal computer).

You must provide another network connection to the Remote ESS for peer-to-peer Remote Copy Web Copy Service support.

ESS Specialist

The ESS includes the ESS Specialist. The ESS Specialist is a Web-based interface that allows you to configure the ESS.

From the Web interface, you can perform the following tasks:

- Monitor error logs
- View and change the configuration:
    - Add or delete SCSI and fibre-channel host systems
    - Configure SCSI and fibre-channel host ports on the ESS
    - Define controller images for System/390
    - Define fixed-block (FB) and count-key-data (CKD) disk groups
    - Add FB and CKD volumes
    - Assign volumes to be accessible to more than one host system
    - Change volume assignments
- Change and view communication resource settings, such as e-mail addresses and telephone numbers
- Authorize user access

With the ESS Specialist, you can view the following information:

- The external connection between a host system and an ESS port
- The internal connection of SCSI ports to cluster 1 or cluster 2
- How the storage space is allocated to FB and CKD volumes

ESS Copy Services

Copy Services provides a Web-based interface for managing peer-to-peer remote copy (PPRC) and FlashCopy™ commands. Copy Services collects information from your IBM storage servers on a single Copy Services server.

Copy Services is part of the IBM Enterprise Storage Server Copy Services Web interface. You access Copy Services from the ESS Specialist main menu.

Use the Copy Services panels to view and define:

- **Volumes**

    The Volumes panel allows you to view volumes and define them as source or target volumes for the PPRC.

- **Server**

    The Server panel allows you to work with logical servers as complete entities. You can build tasks to place the volumes of a logical server within a peer-to-peer relationship with the volumes of another logical server.

    You can also build a task to remove similar groups of volumes from an existing peer-to-peer relationship.

- **Paths**

    The Paths panel displays the current status of paths between one physical server and the servers to which it is connected.

You can also use this panel to add or remove copy service paths.

- **Tasks**

  The Tasks panel allows you to manage tasks you have defined. You can run, remove, export, or import tasks.

- **Configuration**

  The Configuration panel allows you to add to or save the existing configuration. You can also use this panel to display the error log.

ESS Expert

The IBM StorWatch Enterprise Storage Server Expert (ESS Expert) is an optional software product that you can purchase to use with the ESS. The ESS Expert Web interface provides storage-resource management for the IBM storage servers.

- Asset management

  ESS Expert collects and displays asset management data.

- Capacity management

  ESS Expert collects and displays capacity management data.

- Performance management

  ESS Expert collects and displays performance management data. For example:

  – Number of I/O requests
  – Number of bytes transferred
  – Read and write response time
  – Cache usage statistics

ESS Expert allows you to schedule the information collection. With this information, you can make informed decisions about volume placement and capacity planning as well as isolate I/O performance bottlenecks.

Service interface

The ESS provides service interface ports for external connection of a service terminal.

IBM or your service provider can perform service on the ESS by using an IBM mobile service terminal (MoST) or equivalent.

Remote services support

The ESS service interface also provides remote service support with call-home capability for directed maintenance for service personnel.

You provide an analog telephone line to enable this support. The service interface provides an RS232 connection through a modem switch and modem to the analog telephone line.

You need to order a modem and modem switch. The first ESS you order requires this equipment. The modem and modem switch support up to seven ESSs. The cable length from the ESS to the modem switch should be a maximum of 15 m (50 ft).

The ESS provides the following service functions:

- Continuous self-monitoring initiates a call (call home) to service personnel if a failure has occurred. Because service personnel who respond to the call know about the failing component, the service personnel can reduce the repair time.

- Error and problem logs are available that service personnel can access remotely to analyze potential failures.

- Remote support allows the ESS to correct many types of problems. When the ESS reports a problem, service personnel can often create a correction and apply it from the remote location.

Configuration

IBM logically configures your ESS during installation. After IBM installs the ESS, you can perform additional configuration by using the ESS Web interfaces. You can also change the remote service functions.

Fibre-channel connection

The ESS provides fibre-channel connection to open host systems that it supports. Fibre-channel interconnection architecture provides a variety of communication protocols on the ESS. See Figure 3 on page 9. The units that are interconnected are referred to as *nodes*. Each node has one or more ports

An ESS is a node in a fibre-channel network. Each port on an ESS fibre-channel host adapter is a fibre-channel port. A host is also a node in a fibre-channel network. Each port on a host fibre-channel adapter is a fibre-channel port.

Each port attaches to a serial-transmission medium that provides full-duplexed communication with the node at the other end of the medium.

ESS architecture supports three basic interconnection topologies. See Figure 3 on page 9.

Point-to-point
　　　　Point-to-point allows you to interconnect ports directly.

Fabric　　To allow multiple nodes to be interconnected, you can use a fabric that provides the necessary switching functions to support communication between multiple nodes. You can implement a fabric using available vendor products.

arbitrated loop
　　　　Arbitrated loop is a ring topology that allows you to interconnect a set of nodes. The maximum number of ports you can have for a fibre-channel arbitrated loop is 128.

Note: IBM does not support fabric topology for arbitrated loop.

Point to Point

Fabric

```
┌────┬─┐        ┌─┬────┐        ┌────┬─┐   ┌─┐ ┌──────┐
│Node│N├────────┤N│Node│        │Node│N├───┤F│ │Fabric│
└────┴─┘        └─┴────┘        │    ├─┤   ├─┤ │      │
                                │    │N├───┤F│ │      │
                                └────┴─┘   └─┘ │      │
                                              │      │
                                ┌────┬─┐   ┌─┐ │      │
                                │Node│N├───┤F│ │      │
                                └────┴─┘   └─┘ │      │
                                              └──────┘
```

Arbitrated Loop

```
┌────┬──┐
│Node│NL├──┐
│    │  │  │
│    │  │  │
└────┴──┘  │
┌────┬──┐  │
│Node│NL├──┘
│    │  │  │
│    │  │  │
└────┴──┘  │
┌────┬──┐  │
│Node│NL├──┘
│    │  │
│    │  │
└────┴──┘
```
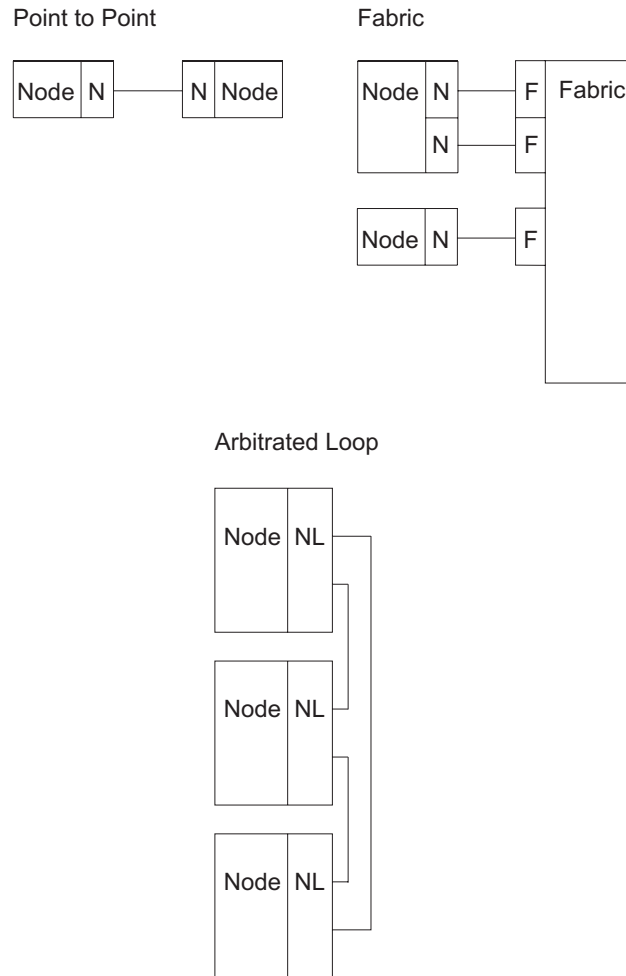
N=Node port   NL=Node loop port

F=Fabric port

Figure 3. Fibre-channel topologies

Fibre-channel identification

Each fibre-channel port on an ESS has a unique fibre-channel port identifier. The format and content of the fibre-channel port identifier is determined by the manufacturer of the link control facility for the applicable fibre-channel port. The identifier is an eight-byte field, which is used in fibre-channel protocols to uniquely identify the fibre-channel port, such as the ESS fibre-channel port.

Each ESS is assigned a unique world-wide node identifier. IBM uses the IEEE registration authority to obtain these identifiers.

The format of the node identifier is determined by the first four bits of the field, which is referred to as the name address authority (NAA). The format used on ESS as provided by the IEEE registration authority is shown in Figure 4.

IBM places a label with the 16 hex-character node identifier on each ESS.

Figure 4. IEEE registered name node identifier format

Bit	Value	Description
63 - 60	X'5'	NAA (IEEE registered name fibre-channel protocol format)
59 - 36	X'xxxxxx'	IEEE organizationally unique identifier (OUI) (IEEE company ID)
35 - 0	X'yyyyyyyyy'	Vendor-specified identifier (under control of company identified by the OUI)

Hosts attaching to the ESS can use node and node-ID identifiers that are in any format. The identifiers must be world-wide unique or at least unique within the domain of the fabrics attached to the ESS.

ESS has a unique node name for each ESS and a unique node name for each port. When you login in FCPH, you provide a node name and a port name. Each follows the IEEE rules. The node name is the same from all ports on the ESS.

Note: ESS does not accept all world wide identifiers. Check the NAA to ensure the following:

- IEEE (0001)

- IEEE extended (0010)

- IEEE registered (0101)

- IEE registered extended (0110)

Fibre-channel classes of service

On fibre-channel attachments, the ESS supports class 2 or 3 operation on a point-to-point or fabric topology. It supports class 3 operation on an arbitrated loop topology.

Fibre-channel multipathing considerations

If you use the peer-to-peer remote copy facility on hosts that communicate with multiple physical links of an ESS, you should configure the physical links of the ESS so that they are balanced between clusters. You should balance the physical links within the cluster between I/O bays.

Introducing the IBM Subsystem Device Driver

The IBM Subsystem Device Driver resides in the host server with the native disk device driver for IBM Enterprise Storage Server (ESS). It uses redundant connections between the host server and disk storage in an ESS to provide enhanced performance and data availability. These connections comprise many different components through which data flows during input and output processes. Redundancy and the ability to switch between these components provides many different paths for the data to travel. In the event of failure in one input-output path, it automatically switches to another input-output path. This automatic switching in the event of failure is called *failover*.
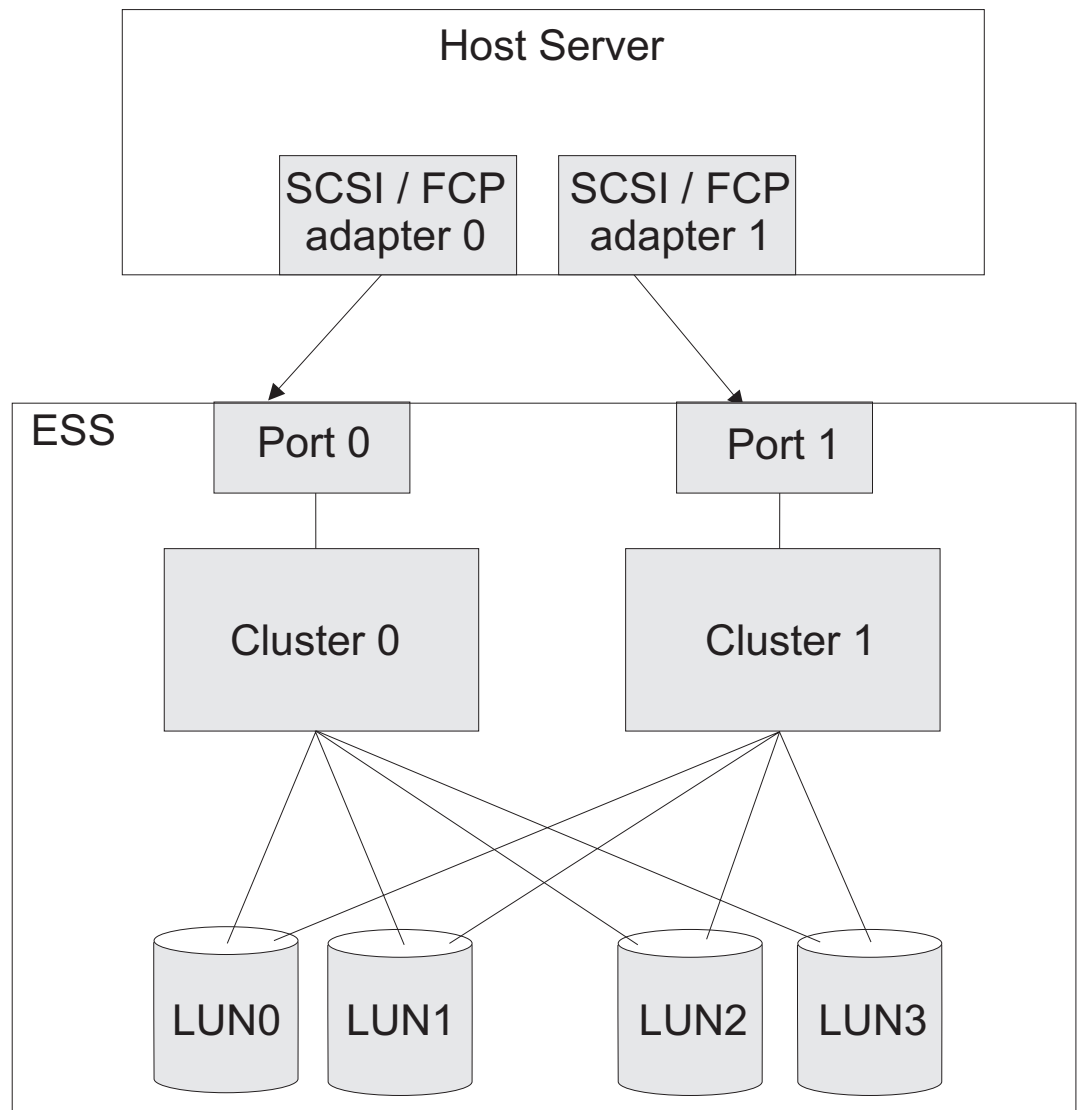
S009000Q

Figure 5. Multi-path connections between host server and ESS LUNs

The IBM Subsystem Device Driver provides the following functions:

- Enhanced data availability
- Automatic path failover and recovery to an alternate path
- Dynamic load balancing of multiple paths
- Setting path selection policies (AIX only)
- Concurrent download of licensed internal code

In most cases, host servers are configured with multiple host adapters with SCSI or fibre-channel connections to an ESS that, in turn, provides internal component redundancy. With dual clusters and multiple host interface adapters, the ESS provides more flexibility in the number of input-output (I/O) paths that are available. When there is a failure, the IBM Subsystem Device Driver reroutes I/O operations from the failed path to the remaining paths. This prevents a bus adapter on the host server, an external SCSI cable, a fiber connector, a cluster on the ESS, or a host interface adapter on the ESS from disrupting data access.

Chapter 1. IBM Enterprise Storage Server overview **11**

In addition, multipath load balancing of data flow prevents a single path from becoming overloaded with I/O.

The IBM Subsystem Device Driver path recovery algorithms have also been improved. The IBM Subsystem Device Driver will now test **Dead** paths that are not manually offline every 30 minutes. If the test is successful, paths will be brought back online sooner than they have been previously. The improved algorithms also detect failing paths earlier.

Concurrent download of licensed internal code is the capability to download and install licensed internal code on a ESS while applications continue to run. During the time when new licensed internal code is being installed in an ESS, the upper interface adapters inside the ESS may not respond to host I/O requests for approximately 30 seconds. The IBM Subsystem Device Driver makes such failures transparent to the host through its path selection and retry algorithms.

Note: For updated and additional information not included in this publication, see the README file on the IBM Subsystem Device Driver compact disk recordable (CDR) or visit the Subsystem Device Driver web site at: **www.ibm.com/storage/support/techsup/swtechsup.nsf/support/sddupdates** "

Chapter 2. Configurations

Multiple ports

Multiple host connection ports are supported.

Multiple SCSI-3 interfaces

A host connection port can be any of these interfaces:

- Parallel SCSI
- Fibre Channel (Arbitrated Loop and Switch)

Parallel SCSI

Multiple targets

Up to 15 targets are surfaced on each port. Each target can be assigned any valid ID.

Multiple logical units

Up to 64 logical units can be surfaced on each target. All logical units defined to a target are assigned to volumes on the same logical subsystem.

Logical units are the unit of sharing. A logical unit may be surfaced on any number of ports. Such logical units are called *shared logical units*. The target and LUN values used to address a shared logical unit on different ports are not related in any way. They can be any valid values for the interface protocols on the ports. Logical units can be shared on ports with dissimilar interface protocols.

A logical unit can have only one target/LUN address on any one port. If initiators wish to share a logical unit on one port, then they must address the logical unit with the same target/LUN values.

The following items are logical unit entities. There is one set of them for a logical unit. The same values are surfaced on all ports that share the logical unit.

- Inquiry data
- Mode parameters
- Block size
- Capacity (number of blocks)

13

Multiple initiators

Up to four initiators are supported on each port. They may have any IDs that are valid for the interface protocol.

Port behaviors

All logical units surfaced on a port must have the same block size.

Only homogeneous host types are supported on a port. For example, an HP UNIX machine cannot be connected to the same port as an RS/6000. **Note:** This restriction is not enforced.

Fibre channel

Single target

Only one target is surfaced on each port.

Multiple logical units

Up to 16384 (16k) logical units can be surfaced on each target. All fixed block volumes configured are surfaced on all FC ports.

The following items are logical unit entities. There is one set of them for a logical unit. The same values are surfaced on all ports that share the logical unit.

- Inquiry data
- Mode parameters
- Block size
- Capacity (number of blocks)

Multiple initiators

Up to 128 initiators are supported on each port. They may have any IDs that are valid for the interface protocol.

ESS supports up to 512 initiators across all ports.

Port behaviors

Port behaviors do not apply to FC ports.

Chapter 3. Features

Power on

When the ESS is powered on or when it executes a self-initiated reset it does the following on all ports:

- Generate Power on Unit Attention conditions for all initiators on all logical units on all ports.

- Wait until the subsystem has completed its IML before responding to selection, accepting commands, or accepting messages.

Invalid LUN

The following commands execute successfully when a LUN is specified which does not exist:

Inquiry   Standard inquiry data is returned regardless of the EVPD value.

Report LUNs

> Parallel SCSI only

Request Sense The returned sense data indicates `Command Aborted - Logical Unit not supported`. The ACA message is ignored if it is associated with a request sense command for an invalid LUN. This is because we never generate a Contingent Allegiance for an invalid LUN.

All other commands fail with Check Condition status. No sense data is set. No Contingent Allegiance condition is generated.

> End of Parallel SCSI only

> Fibre attached (FCP) only

Mode Select or Mode Sense that attempt to read or update mode page 19h are accepted on LUN 0000h only

All other commands fail with Check Condition status. The returned sense data indicates `Command Aborted - Logical Unit not supported`. No Contingent Allegiance condition is generated.

> End of Fibre attached (FCP) only

Not ready

A logical unit becomes not ready when a **Stop Unit** command is issued. In this state, media access commands are not accepted. They are failed with Check Condition Status. The sense data is set to `Not Ready - Logical Unit not ready, initialization command required`.

A logical unit may also be not ready for a brief period of time following a reset. The target responds to selection immediately following the reset, but media access commands are failed with Check Condition Status until the logical unit is ready. The sense data is set to `Not Ready - logical unit is in the process of becoming ready.`

The media access commands are listed here:

- Format Unit
- Read
- Reassign Blocks
- Send Diagnostic
- Skip Read (AS/400 only)
- Skip Write (AS/400 only)
- Test Unit Ready
- Verify
- Write
- Write and Verify
- Write Same

Contents of new logical unit

When a logical unit is created the data for each block is undefined. Reading a block before it is written will cause some arbitrary data and Good Status to be returned. Media errors will not be reported for blocks that are read before being written by the host (unless an actual media error exists).

Controlling initiator access to FB volumes

ESS uses two different method to control initiator access to FB volumes, depending on the interface type.

- For parallel SCSI attach, ESS maps subsystems to targets on SCSI buses and maps the volumes of those subsystems into LUNs on the targets.
- For FCP attached, ESS allows access to volumes from initiators to be controlled via it's port's World Wide Port Name.

Parallel SCSI bus LUN mapping

ESS requires that a port map be defined for each parallel SCSI port configured. This port map defines:

- Which SCSI IDs are initiators
- Which SCSI IDs are targets that ESS responds to
- For each target that ESS responds to the map defines:
    - Which subsystem the target has access to
    - Maps the 64 LUNs of the target into the 256 volumes of the subsystem

Controlling access of FCP attached initiators

Unlike parallel attach, FCP must be able to support a heterogeneous mix of hosts attached through a single port. This is for a number of reasons:

- There are fewer fibre channel ports on a storage facility than parallel SCSI ports.

- The inherent nature of fibre channel is a fabric attachment for point-to-point or in a loop for FCAL.

As a result the initiator type and identification of which volumes may be accessed for a given host must be made know to the storage facility by identifying the initiator rather than the storage facility port. An initiator will be identified via its FCPH Port Name.

The following information is kept for any initiator that attaches to ESS:

- FCPH port name

- Initiator Type (such as AIX, SUN, NT)

- Whether initiator supports the Report LUNs command.

- If the initiator does not support the Report LUNs command, then a map that maps LUNs x'0000' to x'00FF' into 256 volumes.

- If the initiator does support the Report LUNs command, then the identification of which volumes can be accessed by this host.

The following occurs for an initiator logging into ESS whose FCPH port name is not known to the storage facility:

- The host is assumed to be an AIX host.

- The host is assumed to support the Report LUNs command.

- An ODM setting for the storage facility determines whether to allow access to all non AS/400 LUNs or no LUNs for unknown initiators.

Chapter 4. Queuing (task creation)

Both tagged and untagged queuing are supported. The queuing model is that of the SCSI-3 SAM. There is one queue per logical unit. Queue tag message processing, Contingent Allegiance conditions, and reservations are implemented on a logical unit basis and affect the target's decision whether to accept or execute a new command.

Logical unit number (LUN) identification is by the identify message. If no identify message is received prior to command phase then busy is dropped (going Bus Free) and a SCSI protocol error is recognized.

Note: The terms *command, task,* and *I/O process* are synonyms in this chapter.

Note: The terms *Auto Contingent Allegiance condition* and *Contingent Allegiance condition* are synonyms in this chapter.

Untagged queuing

One active command from each initiator is allowed without a queue tag. This command can coexist with commands that were issued with queue tags.

The target may return Busy status if it does not have enough free resources to accept the command. The target will not return Busy status if the initiator has no other outstanding commands for the specified logical unit (unless NACA=0 and another initiator has a Contingent Allegiance condition).

The second concurrent untagged command from an initiator is terminated with Check Condition status. The sense data is set to `Command Aborted - Overlapped Commands Attempted.` All of the initiator's outstanding commands for the logical unit are aborted.

Tagged queuing

A variable number of commands can be issued by an initiator to a logical unit if they are issued with queue tags.

The target can return Queue Full status if it does not have enough free resources to accept the command. The target does not return Queue Full status if the initiator has no other outstanding commands for the specified logical unit.

The initiator must specify the logical unit number with an Identify message. The Identify message must be followed by one of these messages:

Head of Queue Tag (21h) The command may begin execution immediately. Subsequent commands may not begin execution until this command completes.

Ordered Queue Tag (22h) The command begins execution after all previously issued commands complete. Subsequent commands may not begin execution until this command completes (unless they are issued with Head of Queue tag messages).

Simple Queue Tag (20h) This command can begin execution immediately unless an Ordered or Head of Queue tag is in the queue or is executing.

When the target reconnects to an initiator to continue a tagged command, the target sends a Simple Queue Tag message following an Identify message.

Request sense commands get special handling and a higher priority. This special handling can cause request sense commands to violate some of the queue tag rules.

Auto contingent allegiance condition

This section is concerned with commands received from an initiator. There are no such restrictions on messages received from an initiator.

Definitions:

faulting command The command that terminated with Check Condition status (causing the Auto Contingent Allegiance condition).

faulted initiator The initiator that issued the faulting command.

ACA command A command that has the ACA task attribute. It was sent with an ACA message.

The NACA bit is ignored and the ACA state is not kept for commands issued to unconfigured LUNs.

ACA condition (with NACA=1)

If a Contingent Allegiance condition exists for a logical unit and the faulting command had NACA=1, then a new command is accepted and executed if all of these conditions are met:

- The new command is originated by the faulted initiator.

- The new command is sent with an ACA message.

- No other ACA command is active for the logical unit.

If one of these conditions is not met, the new command is failed with ACA Active status or Busy status. The status is Busy if the new command is not from the faulted initiator, and the new command has NACA=0. Otherwise, the status is ACA Active.

The Contingent Allegiance condition is cleared when a Clear ACA message is received from the faulted initiator.

ACA condition (with NACA=0 in the faulting command)

If a Contingent Allegiance condition exists for a logical unit and the faulting command had NACA=0, then a new command is accepted and executed and the Contingent Allegiance condition is cleared if both of these conditions are met:

- The new command is originated by the faulted initiator.

- The new command is not an ACA command.

If the new command is not originated by the faulted initiator, then the new command is failed with ACA Active status or Busy status. The status is Busy if the

new command has NACA=0. Otherwise, the status is ACA Active. The Contingent Allegiance condition is not cleared.

If the new command is an ACA command originated by the faulted initiator, then Check Condition status is returned and the sense data is set to `Illegal Request - Invalid Message`. The Contingent Allegiance condition is cleared and a new one is generated.

No ACA condition

| If a Contingent Allegiance condition does not exist for a configured logical unit and an ACA command is received, then Check Condition status is returned and the sense data is set to `Illegal Request - Invalid Message`.

| ACA tags do not cause Check Conditions on unconfigured LUNs.

Holding off tasks because of resource constraints

When ESS encounters either a long busy or a situation where internal resources are becoming constrained the following occurs:

* If there is a task queued on that LUN for the initiator then Queue Full status is returned.

  Note: This is done without regard for whether the task being queued has a queue message tag.

* If there is not a task queued on that LUN then:

  1. If there is a Unit Attention owed to the initiator from that LUN then the task is accepted and disconnected. The task is thrown away by ESS.

  2. If sufficient resources exists to queue the task but not execute it, queue the task.

  3. Otherwise:

     a. Unit Attention (commands aborted by another initiator) is set as pending for the initiator from this LUN.

     b. The task is accepted and disconnected

        Note: Should disconnect privilege not be granted, then ESS hangs on the bus until sufficient resources exist to execute the task.

     c. The task is thrown away by ESS.

Reservation

This section is concerned with commands received from an initiator. There are no such restrictions on messages received from an initiator. There are two incompatible methods that may be used by an initiator to reserve a LUN.

1. Traditional Reserve

   This uses the Reserve and Release commands to reserve a LUN to a single initiator.

2. Persistent Reserve

This uses Persistent Reserve In and Persistent Reserve Out to manage reservations of a LUN. A key part of Persistent Reserve is an eight byte key registered by an initiator. Reserve and Release commands will not be accepted from an initiator that has registered a key with that LUN.

Reserve / Release

A logical unit is reserved for an initiator only after Good status is returned for a Reserve command. The reservation is in effect until the initiator that issued the Reserve command issues a Release command or until the logical unit is reset.

If a logical unit is reserved when a new command is received, the target examines the command opcode and the ID of the initiator which issued the command. The target uses this information to determine if it will return Reservation Conflict status for that command. The target uses these rules to make this determination:

- If the issuing initiator is the one that made the reservation and also the one to receive the reservation, then all commands are permitted.

- If the issuing initiator is neither the one that made the reservation nor the one to receive the reservation, then:
    - Inquiry is permitted
    - Request Sense is permitted
    - Read capacity is permitted
    - Log Sense is permitted
    - Report LUNs is permitted
    - Start/Stop Unit is permitted, if Start=1
    - Release is permitted but is ignored
    - Any other command results in Reservation Conflict status

- If the issuing initiator is the one that made the reservation but is not the one to receive the reservation, then:
    - Inquiry is permitted
    - Request Sense is permitted
    - Read capacity is permitted
    - Log Sense is permitted
    - Report LUNs is permitted
    - Start/Stop Unit is permitted, if Start=1
    - Reserve is permitted
    - Release is permitted
    - Any other command results in Reservation Conflict status

- If the issuing initiator is not the one that made the reservation but is the one to receive the reservation, then:
    - Reserve results in Reservation Conflict status
    - Release is permitted but is ignored.
    - Any other command is permitted

Persistent reserve

The persistent reservations management method is used among multiple initiators that require operations to be protected across initiator failures, which usually involve hard resets. Persistent reservations persist across recovery actions, to provide initiators with more detailed control over reservations recovery. Persistent reservations for failing initiators may be preempted by another initiator as part of the recovery process. Persistent reservations are retained until:

- They are released, preempted, cleared by a Persistent Reserve Out Command,

- They are cleared when the initiator that holds the reserve unregisters (issues a Persistent Reserve Out command with a service action of Register and a service key other than the key that holds the reserve),

- They are reset by a machine IML (only if APTPL is not set).

Since persistent reservations are not reset by the Target Reset task management function or other global actions, they may be used to enforce device sharing among multiple initiators. The Persistent Reserve Out and Persistent Reserve In commands provide the basic mechanism for dynamic contention resolution in multiple initiator systems using multiple port targets. The identification of persistent reservations using the reservation key makes it possible to determine which ports hold conflicting persistent reservations and to take over persistent reservations from failing or uncooperative initiators. A reservation key is set by an initiator with a Persistent Reserve Out command with a service action of Register. An eight byte reservation key is transmitted in the data of that command.

To reserve a LUN using a persistent reserve the initiation must first register with the LUN, then issue a Persistent Reserve Out command with a service action of Reserve.

If a logical unit is reserved when a new command is received, the target examines the command opcode, the reservation state and whether the initiator that issued the command is registered for the LUN or not. ESS uses this information to determine if it will return Reservation Conflict status for that command.

ESS uses these rules to make this determination:

- If the issuing initiator holds the reservation, then all commands are permitted.

- If the issuing initiator is registered but does not hold the reservation then:
    - Inquiry is permitted
    - Request Sense is permitted
    - Read capacity is permitted
    - Log Sense is permitted
    - Report LUNs is permitted
    - Start/Stop Unit is permitted, if Start=1
    - Release is permitted but is ignored
    - Persistent Reserve In is permitted
    - Persistent Reserve Out is permitted, see command description for Reservation Conflict Status that may be returned

- If the reservation type is Exclusive Access then any other command results in Reservation Conflict status

- If the reservation type is Write Exclusive then read commands are permitted, any other command results in Reservation Conflict status

- If the reservation type is Write Exclusive, Registrants Only or Exclusive access, Registrants Only then all commands are permitted

• If the issuing initiator is not registered on this LUN then:

- Inquiry is permitted

- Request Sense is permitted

- Read capacity is permitted

- Log Sense is permitted

- Report LUNs is permitted

- Start/Stop Unit is permitted, if Start=1

- Release is permitted but is ignored

- Persistent Reserve In is permitted

- Persistent Reserve Out is permitted, see command description for Reservation Conflict Status that may be returned

- If the reservation type is Exclusive Access or Exclusive Access, Registrants Only then any other command results in Reservation Conflict status

- If the reservation type is Write Exclusive or Write Exclusive, Registrants Only then read commands are permitted, any other command results in Reservation Conflict status

Format in progress

If a logical unit is executing a format operation when a new command is received, the target examines the command opcode to determine if it will execute the new command.

Inquiry Execute the command.

Request Sense Execute the command, return sense data indicating `Not Ready - format in progress.` The Sense Key Specific Bytes will indicate the progress of the format operation.

All Other Commands

Return Check Condition status. A Contingent Allegiance condition is created.

Unit attention condition

If the target has a Unit Attention condition to report to an initiator and a new command is received from that initiator, then the target examines the command opcode to determine if it executes the new command.

Inquiry Execute the command, and preserve the Unit Attention condition.

Request Sense
>Execute the command, return the sense data associated with the Unit Attention condition, and clear the Unit Attention condition.

All Other Commands
>Return Check Condition status. A Contingent Allegiance condition is created. The sense data is set to the sense data that is associated with the Unit Attention condition. The Unit Attention condition is cleared.

Multiple port issues

A logical unit can be surfaced on more than one port. In this configuration, there is still one queue per logical unit. Such a logical unit is called a *shared logical unit*. Task management operations, Ordered Queue tags, Head of Queue tags, reservations, and Contingent Allegiance conditions affect all initiators that attempt to send commands to the logical unit regardless of which port they use to connect.

Reservation

A logical unit can only be reserved for one initiator on one port. It can only be released by the initiator that issued the Reserve command on the same port on which the Reserve command was issued.

A reservation ends when the logical unit or one of the targets that surfaces the logical unit is reset. The reset does not have to be received on the same port as the Reserve command was received.

Third party reservation can only be done for an initiator that connects using the same port as the initiator that issues the Reserve command.

Contingent allegiance

A logical unit can only have a Contingent Allegiance condition for one initiator at a time. No Check Conditions are issued on other ports or to initiators on the same port until the Contingent Allegiance condition is cleared.

A Contingent Allegiance condition affects all accesses to the logical unit (including those from ports other than the one used by the faulted initiator).

The Contingent Allegiance condition can only be cleared by the initiator that received the Check Condition status on the same port on which the Check Condition status was returned.

A Contingent Allegiance condition ends when the logical unit or one of the targets that surfaces the logical unit is reset. The reset does not have to be received on the same port as the one on which the Check Condition status was returned.

Queued tasks

Tasks received on different ports for the same logical unit are merged into one logical unit queue. Ordering of tasks due to Ordered Queue tags and Head of Queue tags is implemented in this queue.

Checking for overlapped tasks is done on a port basis. Two tasks in the queue may have the same initiator, LUN, and Queue tag values if the commands were received on different ports.

Task management operations

Task management operations affect the whole logical unit queue, not just the commands from the port on which the operation is received.

Note: For this discussion, commands aborted due to the clearing of a Contingent Allegiance condition are treated the same as if an explicit task management operation caused the abort operation.

It is possible for a task management operation to cause a command to be aborted which is active on a different port than the port on which the message is received. In this case, the connection is allowed to end gracefully.

Recommended command timeouts

Regardless of the values listed below, no initiator should time a command for less than 30 seconds unless that initiator supports either Long Busy or polls for a non-responding LUN to resume I/O. This is because of the time it can take ESS to recover from a warm start, failback or failover.

Unless listed below, the recommended initiator timeout value is 30 seconds.

DSO, Read Message 6 minutes

Inquiry   20 seconds

Format Unit, with immed = 0.   6 minutes per GB

Request Sense   20 seconds

Log sense   20 seconds

Start Unit   10 seconds

Test Unit Ready   20 seconds

Write Same   6 minutes per GB, however never less than 30 seconds

Chapter 5. Task management operations

These operations are invoked by initiators to create or abort commands.

Tasks are created when an initiator sends a command to a logical unit. The command may be given a Queue tag.

Task management operations are always accepted from any initiator, even if a reservation or contingent allegiance condition is in effect.

Parallel SCSI invokes these operations using messages. See Figure 73 on page 80 for the message codes to invoke these operations.

Fibre channel has these operations imbedded within the protocol, see Figure 138 on page 129.

Abort Task

This message aborts the current command. The current command is the one last identified by an Identify and Queue tag message by either the initiator or target. The current connection is terminated.

If no logical unit or Queue tag is specified, or if an initiator specifies a Queue tag which is not active, then the current connection is terminated but no other commands are affected.

Abort Task Set

This message aborts all commands for the identified logical unit which were sent by the initiator that sent this message. The current connection is terminated after the commands have been aborted.

If no logical unit is identified, then the current connection is terminated but no other commands are affected.

Clear ACA

Clear ACA clears a Contingent Allegiance condition if it is sent by an initiator that has a Contingent Allegiance condition. Otherwise it is ignored.

When the Contingent Allegiance condition is cleared, the target either resumes pending commands for a logical unit or it aborts them. The action is determined by the QErr and QEMC mode parameters. See "Mode Pages" on page 42 for a description of these parameters.

Clear Task Set

Clear Task Set aborts all commands for the identified logical unit. A Unit Attention is generated for all initiators with active commands other than the one which sent the message. The Unit Attention indicates `Commands Cleared By Another initiator`.

LUN Reset

This message aborts all commands for the identified logical unit. A Reset Unit Attention is generated for all initiators for this logical unit. No port specific information is reset (such as synchronous or wide negotiated interface parameters). A reservation for the logical unit is ended by this message. A Contingent Allegiance condition for the logical unit is ended by this message.

If no logical unit is identified, then the message is ignored.

Target Reset

This message causes a LUN Reset to be executed for all logical units that are surfaced on the target. Port specific information for the target is reset to the default power on state.

Process Logout

Whenever a process is logged out, implicitly or explicitly, the following occurs:

- All commands sent by the initiator being logged out are aborted.
- All reservations held by the initiator being logged out are reset.

  Note: Persistent reservations are not affected by implicit logouts.

- Any contingent allegiance to the initiator is cleared.

This reset does not reset mode pages.

Bus Reset or Link Failure

A SCSI Bus Reset or a Fibre Channel Link Failure causes all initiators on the port to be reset in the manner defined in "Process Logout."

Chapter 6. Commands

This chapter defines the commands that the target supports.

All fields are *big endian*. If a field is larger than one byte, the most significant byte occupies the lowest numbered byte of the field.

Common fields

Some fields are common to many commands. They are defined here.

Operation Code

Figure 6. Operation Code

Group Code Characteristics	Op code	Definition
'000'b 6 byte commands	X'00'	"Test Unit Ready" on page 70
	X'03'	"Request Sense" on page 62
	X'04'	"Format Unit" on page 32
	X'07'	"Reassign Blocks" on page 60
	X'08'	"Read" on page 58
	X'0A'	Write, see "Write, Write and Verify" on page 71
	X'12'	"Inquiry" on page 33
	X'15'	"Mode Select" on page 40
	X'16'	"Reserve" on page 64
	X'17'	"Release" on page 63
	X'1A'	"Mode Sense" on page 50
	X'1B'	"Start/Stop Unit" on page 69
	X'1D'	"Send Diagnostic" on page 65
'001'b 10 byte commands	X'25'	"Read Capacity" on page 60
	X'28'	"Read" on page 58
	X'2A'	Write, see "Write, Write and Verify" on page 71
	X'2E'	Write and Verify, see "Write, Write and Verify" on page 71
	X'2F'	"Verify" on page 70
	X'35'	"Synchronize Cache" on page 69
	X'3B'	"Write Buffer" on page 72
	X'3C'	"Read Buffer" on page 59
'010'b 10 byte commands	X'41'	"Write Same" on page 73
	X'4D'	"Log Sense" on page 37
	X'55'	"Mode Select" on page 40
	X'56'	"Reserve" on page 64
	X'57'	"Release" on page 63
	X'5A'	"Mode Sense" on page 50
	X'5E'	"Persistent Reserve In" on page 52
	X'5F'	"Persistent Reserve Out" on page 54
'011'b Reserved		Should an opcode with a group code of 011 be received then ESS attempts to receive a 10 byte CDB and then go bus free.
'100'b 16 Byte commands		
'101'b 12 Byte commands	X'A0'	"Report LUNs" on page 61
'110'b Vendor Specific commands		
'111'b Vendor Specific commands	X'E8'	"Skip Read" on page 66
	X'EA'	"Skip Write" on page 67

Allocation Length

This field is used in commands which cause a Data In action to occur. The target does not send more bytes using Data In than is specified by the allocation length.

The target can return less data than is specified by the allocation length.

If the target has more data to return than the Allocation Length permits, then the target truncates the data after sending the number of bytes specified by the Allocation Length.

NACA (Normal Auto Contingent Allegiance)

This bit determines whether the SCSI-2 Contingent Allegiance rules or the SCSI-3 Auto Contingent Allegiance rules are used when a command terminates with Check Condition status.

In either case, all non-ACA tasks in a logical unit with an Auto Contingent Allegiance condition are put in the blocked state. These tasks cannot become a current task. The faulted initiator is the one that received the Check Condition.

Initiators other than the faulted initiator receive Busy status if they send commands with NACA=0 to the logical unit. They receive ACA Active status if they send commands with NACA=1.

NACA=0 (in the faulted command) The ACA condition is cleared when the faulted initiator sends any command. The state of the other tasks in the logical unit at this time is determined by the value of the Qerr bit. See "Mode Select" on page 40.

NACA=1 (in the faulted command) The faulted initiator can send one command to the logical unit if that command has the ACA attribute. This command is executed. The ACA is not cleared.

The ACA condition is cleared when the faulted initiator does a Clear ACA task management function. The state of the other tasks in the logical unit at this time is determined by the value of the Qerr bit. See "Mode Select" on page 40.

The NACA bit is ignored and the ACA state is not kept for commands issued to unconfigured LUNs.

Page Length or Parameter List Length

For pages of data returned to the initiator using Data In, it is necessary to indicate how many bytes comprise the data. A Page Length or Parameter List Length field is included in the page definition for this purpose. The field indicates how many bytes follow the field.

The value of the Length field is unaffected by the Allocation Length specified by the initiator.

Commands which cause a Data Out to occur (other than block write operations) have a Parameter List Length field in the Command. This field tells the target how many bytes to transfer using Data Out for the command. It is generally not an error for an initiator to set this field to zero.

Logical Block Address The beginning block for a media access command.

Transfer Length　　The number of blocks that are to be transferred for a media access command. It is generally not an error for an initiator to specify Transfer Length=0.

Command verification

The target performs the following verifications on the Command Descriptor Block (CDB) of commands sent to it.

If any of the named fields contain invalid values, Check Condition status is returned. The sense data indicates `Illegal Request-Invalid Field in CDB.`

For commands that address logical blocks, the target verifies that all addressed logical blocks are within the capacity of the logical unit. If not, Check Condition status is returned. The sense data indicates `Illegal Request - Logical Block Address Out Of Range.`

If the Operation Code is not one of the supported commands, then Check Condition status is returned. The sense data indicates `Illegal Request - Invalid Command Operation Code.`

Note:　The target does not verify that reserved fields in the CDB contain zeros.

Format Unit

Figure 7. Format Unit

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'04'							
1	Reserved			FmtDat	Ignored			
2	0							
3-4	Ignored							
5	0		Reserved			NACA	0	

The Format Unit command writes zeros to all sectors of the logical unit.

If FmtDat is one, then the target receives a defect list using data out. See Figure 8 "Format of Defect List".

Figure 8. Format of defect list

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Reserved							
1	0						Immed	0
2-3	Defect List Length							
4-n	Ignored							

Immed　　Immed=0 specifies that the target returns status after the format operation is finished.

Immed=1 specifies that the target returns Good status immediately after the command begins execution.

Defect List Length This field indicates how many bytes follow byte 3. If the defect list length is greater than 512, then the command is rejected as an invalid request with invalid parameter in the parameter list.

Inquiry

Figure 9. Inquiry

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'12'							
1	Reserved						0	EVPD
2	Page Code							
3	Reserved							
4	Allocation length							
5	0		Reserved			NACA	0	

The Inquiry command requests that information about the logical unit be sent to the initiator using Data In.

The Inquiry command will neither present Unit Attention status if it is pending, nor will it reset that status.

Enable Vital Product Data (EVPD)=0 specifies that standard inquiry data be returned. The page code must also be zero. If EVPD is zero and page code is not zero, then the target returns Check Condition status. The sense data indicates `Illegal Request-Invalid Field In CDB`.

EVPD=1 specifies that vital product data be sent to the initiator. The Page Code field specifies which page of VPD is to be sent. If the page code specifies a page which is not supported, the target returns Check Condition status. The sense data indicates `Illegal Request-Invalid Field In CDB`.

Inquiry Data - EVPD=0

Figure 10. (RS/6000) Inquiry Data - EVPD=0

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Peripheral Qualifier=0			Peripheral Device Type=0				
1	RMB=0	Reserved						
2	ISO=0		ECMA=0			ANSI=3		
3	Reserved	TrmTsk=0	NACA=1	Reserved	Response data format=2			
4	Additional length=159 (X'9F')							
5	Reserved							
Data returned in bytes 6-7 for parallel SCSI initiators:								
6	0			MultiP	Mchngr=0	ACKQREQQ =0	Addr32=0	Addr16=1
7	RelAdr=0	WBus32=0	WBus16=1	Sync=1	Linked=0	TranDis=0	CmdQue=1	SftRe=0
Data returned in bytes 6-7 for FCP initiators:								
6	0			MultiP	Mchngr=0	ACKQREQQ =0	0	
7	RelAdr=0	0			Linked=0	TranDis=0	CmdQue=1	SftRe=0
8-15	ASCII Vendor ID='IBM'							
16-19	ASCII Product Type='2105'							
20-22	ASCII Model Number=							
	'E10'	Single Phase Power						
	'E20'	Three Phase Power						
	'F10'	Single Phase Power, E10 Follow-on						
	'F20'	Three Phase Power, E20 Follow-on						
23-31	spaces (20h)							
32-35	ASCII Microcode EC Level							
36-43	ASCII Unit Serial Number							
44-55	spaces (20h)							
56-95	Reserved							
96-97	Interface ID							
	Bits	**Definition**						
	15-8	Zero						
	7	Cluster						
	6	Zero						
	5	I/O Board						
	4	Zero						
	3-2	Card Slot						
	1	Zero						
	0	Port						
98-101	ASCII Plant of Manufacture							
102-106	ASCII Date of Manufacture							
107-163	0							

Peripheral Qualifier / Peripheral Device Type

If the Inquiry command is directed at a LUN that does not exist in the target, then the Inquiry data has Qualifier=1h and Peripheral Device Type=1Fh. This indicates that the specified LUN is not installed.

	If the Inquiry command is directed at a LUN that does exist in the target, then the Inquiry data has Qualifier=0h and Peripheral Device Type=0h. This indicates that the logical unit is present.
RMB=0	The logical unit is not removable.
ISO=0	The target does not claim compliance to the International Organization for Standards (ISO) version of SCSI (ISO DIS 9316).
ECMA=0	The target does not claim compliance to the European Computer Manufacturers Association (ECMA) version of SCSI (ECMA-111).
ANSI=3	The target supports American National Standards Institute (ANSI) SCSI version 3.
TrmTsk=0	The target does not support the Terminate I/O Process message.
NACA=1	The target supports the NACA bit in the Command Descriptor block.

Response Data Format=2 This Inquiry page is in the format defined in ANSI Small Computer System Interface-3 (SCSI-3).

MultiP	The target supports multi-port access to LUNs. This bit is set to one if any of the following is true:

- The volume is configured to more than one SCSI port

- The volume is configured to more than one fibre channel initiator

- The volume is configured at least one SCSI port and at least one fibre channel initiator.

MChngr=0	The target is not part of a media changer.
ACKQREQQ=0	The target does not support request or acknowledge data transfer on a Q cable.
Addr32=0	The target does not support 32-bit wide SCSI addressing.
Addr16=1	The target supports 16-bit wide SCSI addressing.
RelAdr=0	The target does not support relative addressing mode.
WBus32=0	The target does not support 32-bit wide data transfer.
WBus16=1	The target supports 16-bit wide data transfer.
Sync=1	The target supports synchronous data transfer.
Linked=0	The target does not support linked commands.
TranDis=0	The target supports neither the Continue Task nor the Target Transfer Disable message.
CmdQue=1	The target supports tagged command queuing.
ASCII fields	These fields are left-aligned and padded with space characters (20h).

Inquiry Data - EVPD=1 - Page 0h

Figure 11. Inquiry Data - Page 0h

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Peripheral Qualifier=0			Peripheral Device Type=0				
1	Page code=0							
2	Reserved							
3	Page length=3							
4	X'00' (Supported Inquiry page)							
5	X'80' (Supported Inquiry page)							
6	X'83' (Supported Inquiry page)							

This page lists the supported Inquiry page codes.

Qualifier / Peripheral Device Type Inquiry data will have Qualifier=0h and Peripheral Device Type=0h (disk). This indicates that the logical unit is present.

Inquiry Data - EVPD=1 - Page 80h

Figure 12. Inquiry Data - Page X'80'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Peripheral Qualifier=0			Peripheral Device Type=0				
1	Page code=X'80'							
2	Reserved							
3	Page length=8							
4-11	Unit Serial Number							

Qualifier / Peripheral Device Type Inquiry data will have Qualifier=0h and Peripheral Device Type=0h (disk). This indicates that the logical unit is present.

Inquiry Data - EVPD=1 - Page 83h

Figure 13. Inquiry Data - Page X'83'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Peripheral Qualifier=0			Peripheral Device Type=0				
1	Page code=X'83'							
2	Reserved							
3	Page length=36							
4	Reserved				Code Set=2			
5	Reserved				Identifier Type=1			
6	Reserved							
7	Identifier length=32							
8-15	Vendor ID='IBM'							
16-31	Product ID							
32-39	Unit Serial Number							

This page contains an identifier (bytes 8-39) for the logical unit that is unique. A system can use this identifier to determine if two paths exist to the same logical unit.

Qualifier / Peripheral Device Type Inquiry data will have Qualifier=0h and Peripheral Device Type=0h (disk). This indicates that the logical unit is present.

Code Set A value of 2 indicates that the identifier contains ASCII data.

Identifier Type A value of 1 indicates that the first eight bytes of the identifier is a vendor ID. The vendor is responsible for ensuring the uniqueness of the subsequent bytes of the identifier.

Log Sense

Figure 14. Log Sense

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'4D'							
1	Reserved						0	
2	B'01'		Page code					
3-4	Reserved							
5-6	0							
7-8	Allocation length							
9	0		Reserved			NACA	0	

The Log Sense command requests that device statistics be returned to the initiator using Data In.

Page Code specifies which page is returned. The supported page codes are listed in Figure 15, "Log Sense Page X'00'".

Figure 15. Log Sense Page x'00'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page code=0					
1	0							
2-3	Page Length=3							
4	First supported page code=0							
5	Second supported page code=x'30'							
6	Third supported page code=x'31'							

Log Sense Page 30

Log Sense Page 30h returns performance counter information. See Figure 16 on page 38, "Log Sense Page X'30'".

The counters for page 30h are maintained by each LUN. Issuing a Log Sense command to a LUN causes the statistics for that LUN to be returned.

The statistics are measured at the subsystem controller. No attempt is made to extract statistics from underlying devices.

All statistics are kept in counters that individually rollover when overflowed.

All counters are unsigned integers.

Figure 16. Log Sense Page X'30'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page code=x'30'					
1	0							
2-3	Page Length=80 (x'50')							
4-6	0							
7	Page list length=76 (X'4C')							
8-23	0							
24-25	Overruns							
26-27	Underruns							
28-31	Cache Read Hits							
32-39	0							
40-43	Cache Fast Writes							
44-51	0							
52-55	Read Operations							
56-59	Write Operations							
60-75	0							
76-79	Idle Loop Count							
80-81	Idle Loop Period = 100 (10.0 microseconds)							
82-83	0							

Overrun This counter is incremented when a write operation is unable to receive data from the initiator.

Underrun This counter is incremented when a read operation is unable to send data to the initiator because the next data to be sent is not in the target's buffer.

Cache Read Hits This counter is incremented when all read data is returned by the cache immediately following the beginning of the read operation. If the cache return code indicates that it must go to a disk to get the data, then this counter is not incremented.

Cache Fast Writes This counter is incremented when free buffers are returned by the cache immediately following the beginning of the write operation. If the cache return code indicates that the operation must wait for available buffers, then this counter is not incremented.

Read Operations This counter is incremented once for each read operation.

Write Operations This counter is incremented once for each write operation.

Idle Loop Count This counter is incremented once every time the Scan Loop completes without finding work for an I/O operation.

Log Sense Page 31

Log Sense Page 31h returns caching statistics. See Figure 17, "Log Sense Page X'31'".

Figure 17 (Page 1 of 2). Log Sense Page X'31'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page code=x'31'					
1	0							
2-3	Page Length=192 (x'C0')							
4	Format of data returned = 0							
5	ID of volume within subsystem							
6-7	Subsystem number							
8	PIN	NVS	REBLD	Reserved				
9	Reserved							
10-11	ID of the RAID rank that the volume is stored in.							
12-15	Read Normal I/O Requests							
16-19	Read Normal I/O Request Hits							
20-23	Normal Write I/O Requests							
24-27	Normal Write I/O Request Hits							
28-31	Read Sequential I/O Requests							
32-35	Read Sequential I/O Request Hits							
36-39	Write Sequential I/O Requests							
40-43	Write Sequential I/O Request Hits							
44-47	Sequential DASD to Cache Transfer Operations							
48-51	DASD to Cache Transfer Operation Count							
52-55	Cache to DASD Transfer Operation Count							
56-59	DASD Fast Write Operations Delayed Due to NVS Space Constraints							
60-63	Number of record cache Read Misses							
64-67	Count of the number of operations that did not promote a track because free segments were not available.							
68-71	Record Mode Read Operations							
72-75	NVS Space Allocations							
76-79	Operations Delayed Due To Cache Space Constraints							
80-91	Reserved							
92	Non-volume information contained in the record							
	Value	Definition						
	0	No Additional Information						
	1	Home Area RAID rank Information						
	2-FF	Not Defined						
93	Device Adapter ID							
Values of 94-123 if byte 92=1 (Home Area RAID rank Information):								
94-95	RAID rank ID							
96	Number of HDDs in RAID rank							
97	0							
98-99	HDD Sector Size							
100-103	RAID rank Read Requests							
104-107	RAID rank Write Requests							
108-111	RAID rank Fixed Block Sectors Read							

Figure 17 (Page 2 of 2). Log Sense Page X'31'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
112-115	RAID rank Fixed Block Sectors Written							
116-119	RAID rank Read Response Time							
120-123	RAID rank Write Response Time							
124-131	ASCII Serial Number							
132-195	Reserved							

PIN   When PIN=1, there is data pinned for this LUN.

NVS   When NVS=1, data is trapped in a broken NVS for this LUN.

REBLD   When REBLD=1, volume is part of a RAID rank that is being rebuilt.

ASCII Serial Number This is the serial number reported in the Inquiry command bytes 36-43. See Figure 10 on page 33.

Mode Select

Figure 18. Mode Select

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'15'							
1	Reserved			Ignored	Reserved			SP
2-3	Reserved							
4	Parameter List length							
5	0		Reserved			NACA	0	

Figure 19. Mode Select (10 byte)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'55'							
1	Reserved			Ignored	Reserved			SP
2-6	Reserved							
7-8	Parameter List length							
9	0		Reserved			NACA	0	

The Mode Select command allows an initiator to specify logical unit parameters. There is one set of parameters for each logical unit. The parameters are shared for all initiators on all ports that access the LUN.

SP (Save Parameters) SP=0 causes the current parameters to be changed, but not the saved parameters. The existing saved parameters become the current parameters following the next reset of the logical unit.

SP=1 causes both the current and the saved parameters to be changed.

If the parameter list length is not zero, then mode parameters are received using Data Out.

If a page with an unsupported Page Code is received, Check Condition status is returned. The sense data indicates `Illegal Request - Invalid Field in Parameter List`.

If a page with an incorrect page length is received, Check Condition status is returned. The sense data indicates `Illegal Request - Invalid Field in Parameter List`.

If the data received is truncated such that the header, the block descriptor, or the last page is not complete, Check Condition status is returned. The sense data indicates `Illegal Request - Parameter List Length Error`.

If a field in a page is set to an invalid value, Check Condition status is returned. The sense data indicates `Illegal Request - Invalid Field in Parameter List`.

If the specified value for an unchangeable field does not equal the current value for that field, Check Condition status is returned. The sense data indicates `Illegal Request - Invalid Field in Parameter List`.

If the Mode Select command fails with Check Condition status, the target does not change any mode parameters for the logical unit.

If the Mode Select command succeeds with Good status, all targets (on all ports) which surface the logical unit generates unit attentions for the logical unit that was Mode Selected. All initiators receive unit attentions except the one which issued the Mode Select command. The Additional Sense Code will be Mode Parameters Changed.

Mode Select/Mode Sense Parameters

The parameter list must begin with a header as shown in Figure 20 for 6 byte Mode Select/Mode Sense commands and Figure 21 for 10 byte Mode Select/Mode Sense commands. The header is optionally followed by a block descriptor as shown in Figure 22 on page 42. The mode pages follow.

Any combination of valid pages can be sent. Whole pages must be sent. If multiple valid pages with the same page code are sent, the last one is used.

Figure 20. Mode Header for 6 byte Mode Select/Mode Sense commands

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Parameter List length							
1-2	0							
3	Block Descriptor Length							

Figure 21. Mode Header for 10 byte Mode Select/Mode Sense commands

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0-1	Parameter List length							
2-3	0							
4-5	Reserved							
6-7	Block Descriptor Length							

| **Parameter List Length** For Mode Select this field is ignored. For Mode Sense 6
| it is 3 + Block Descriptor Length + the size of all mode
| pages present. For Mode Sense 10 it is 6 + Block
| Descriptor Length + the size of all mode pages present.

Block Descriptor Length 0 if no block descriptor follows.

8 if a block descriptor follows the header.

Figure 22. Block Descriptor

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0-3	Number of Blocks							
4	0							
5-7	Block Length							

Number of Blocks The value in this field is the number of blocks that exist in the
logical unit.

This field must be either zero or the value returned in the block
descriptor by a Mode Sense command for this logical unit. If the field
contains any other number, Check Condition status is returned. The
sense data indicates `Illegal Request - Invalid Field in Parameter
List`.

Block Length The value in this field is the number of bytes in each block.

This field must be the value returned in the block descriptor by a Mode
Sense command for this logical unit . If the field contains any other
number, Check condition status is returned. The sense data indicates
`Illegal Request - Invalid Field in Parameter List`.

Mode Pages

These are the supported mode pages.

Vendor Unique Parameters - Page x'00'

Figure 23. Mode Page x'00'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page Code=x'00'					
1	Page Length=10 (x'0A')							
2	0			QEMC	0			
3-11	0							

QEMC (Queue Error Management Control) This parameter modifies the meaning
of the QErr parameter. See "Control Parameters - Page x'0A'" on
page 47. If QErr=0, then QEMC is ignored. If QErr=1, then QEMC
determines which initiators' commands are aborted when a Contingent
Allegiance condition is cleared.

For QEMC=0, the outstanding commands originated by the faulted
initiator are aborted. The faulted initiator is the one which received the
Check condition. This is the default setting.

For QEMC=1, all outstanding commands for the logical unit are aborted.
This includes all initiators on all ports.

Read/Write Error Recovery Parameters - Page x'01'

Figure 24. Mode Page x'01'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page Code=x'01'					
1	Page Length=10 (x'0A')							
2	AWRE=1	ARRE=1	TB=0	RC=0	EER=0	PER=0	DTE=0	DCR=0
3	Read retry count=0							
4	Correction span=0							
5	Head offset count=0							
6	Data Strobe offset count=0							
7	0							
8	Write retry count=0							
9	0							
10-11	Recovery time limit=0							

AWRE -automatic write reallocation enable This parameter must be set to 1. Sector reallocation is done by the device as needed during write operations.

ARRE -automatic read reallocation enable This parameter must be set to 1. Sector reallocation is done by the target as needed during read operations.

TB -transfer block This parameter must be set to 0. The target does not transfer data for a read operation when it is unable to read the data from the media. In this case, Check condition status is returned without the data being transferred.

> **Note:** It is possible for the target to detect an error during the data transmission for a read operation. In this case, incorrect data can be transferred before the Check condition is returned.

RC -read continuous This parameter must be set to 0. The target does not disable or abridge error recovery procedures. The continuous flow of data can stall when error recovery procedures are run.

EER -enable early recovery This parameter must be set to 0. The target does not attempt to use a quick error recovery procedure before using the full error recovery procedure.

PER -post error This parameter must be set to 0. The target does not report recovered errors.

DTE -disable transfer on error This parameter must be set to 0. The target does not terminate data transfer when it recovers from a media error.

DCR -disable correction This parameter must be 0. The target uses error correction codes when needed.

Disconnect/Reconnect Parameters - Page x'02'

Figure 25. Mode Page x'02'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page Code=x'02'					
1	Page Length=14 (x'0E')							
2	Read buffer full ratio							
3	Write buffer empty ratio							
4-5	Bus inactivity limit=0							
6-7	Disconnect time limit=0							
8-9	Connect time limit=0							
10-11	Maximum burst size							
12	EMDP	FARd=0	FAWrt=0	FAStat=0	Dimm	DTDC=0		
13	0							
14-15	First burst size							

This mode page is maintained separately for parallel SCSI and FCP attached initiators.

Read Buffer Full Ratio   This parameter is ignored. Default setting is zero.

Write Buffer Empty Ratio This parameter is ignored. Default setting is zero.

Maximum Burst Size   On connections following the one in which the command is received, the target does not transfer more data than is specified by this parameter. For fibre channel this specifies the maximum number of blocks in a single sequence. A value of 0 specifies that there is no limit. The units of this parameter are blocks. The number of bytes per block can be determined with a read capacity command.

Parallel SCSI

Default setting is zero.

--- End of Parallel SCSI ---

Fibre channel

The value can be 1 - 64. The default is 64.

--- End of Fibre channel ---

EMDP -enable Modify Data Pointers Setting EMDP=1 allows the target to transfer data out of order. Setting EMDP=0 disallows the target from transferring data out of order. This is the default setting.

FARd -fair arbitration for Reads

FAWrt -fair arbitration for Writes

FAStat - fair arbitration for Status and Messages These parameters must be set to 0. The target does not support fair arbitration. Default setting is zero.

DImm Setting DImm=0 allows the target to transfer data during the same connection in which the command is received.

Setting DImm=1 disallows the target from transferring data during the same connection in which the command is received.

Note: DImm is ignored if disconnect privilege is not granted to the target. See "Message Out phase" on page 80 for a description of the identify message.

First Burst Size

Parallel SCSI

The target does not transfer more data than this for a read or write operation during the connection in which the command is received. A value of 0 specifies that the maximum burst size is used on the initial connection. The units of this parameter are blocks. The number of bytes per block can be determined with a read capacity command.

The default for this is 0.

End of Parallel SCSI

FCP

The initiator does not transfer more data than this on a write until receiving a data request from the target. The units of this parameter are blocks. The number of bytes per block can be determined with a read capacity command.

The value can be 1 - 64. The default is 64.

End of FCP

Format Device Parameters - Page x'03'

Figure 26. Mode Page x'03'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page Code=x'03'					
1	Page Length=22 (x'16')							
2-3	Tracks per zone = 0							
4-5	Alternate sectors per zone = 0							
6-7	Alternate tracks per zone = 0							
8-9	Alternate tracks per logical unit = 0							
10-11	Sectors per track = 64 for volumes ≤ 32 GB, 256 for volumes ≥ 32 GB.							
12-13	Data bytes per physical sector = 512							
14-15	Interleave = 0							
16-17	Track skew factor = 0							
18-19	Cylinder skew factor = 0							
20	SSEC=0	HSEC=1	RMB=0	SURF=0	0			
21-23	0							

The values for these parameters are not changeable. They must be equal to the values returned by the Mode Sense command for this page.

All parameters on this page are ignored by the target.

Rigid Disk Drive Geometry Parameters - Page x'04'

Figure 27. Mode Page x'04'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page Code=x'04'					
1	Page Length=22 (x'16')							
2-4	Number of cylinders = number of tracks allocated to volume divided by the value in byte 5.							
5	Number of heads = 30 for volumes ≤ 32 GB, 64 for volumes ≥ 32 GB							
6-8	Starting cylinder - Write precompensation, same as bytes 2-4							
9-11	Starting cylinder - Reduced write current, same as bytes 2-4							
12-13	Drive step rate = 0							
14-16	Landing zone cylinder = 0							
17	0							RPL = 0
18	Rotational offset = 0							
19	0							
20-21	Medium rotation rate = rotation rate of DDMs in RAID rank							
22-23	0							

The values for these parameters are not changeable. They must be equal to the values returned by the Mode Sense command for this page.

All parameters on this page are ignored by the target.

Cache Parameters - Page x'08'

Figure 28. Mode Page x'08'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page Code=x'08'					
1	Page Length=18 (x'12')							
2	IC=0	ABPF=0	CAP=0	DISC=0	SIZE=0	WCE=1	MF=0	RCD=0
3	Demand read retention priority=0				Write retention priority=0			
4-5	Disable prefetch transfer length=FFFFh							
6-7	Minimum prefetch=0							
8-9	Maximum prefetch=16							
10-11	Maximum prefetch ceiling=16							
12	FSW=0	LBCSS=0	DRA=0	0				
13	Number of cache segments=1							
14-15	Cache segment size=FFFFh							
16	0							
17-19	Non-cache segment size=0							

All named parameters on this page are changeable. Any values are accepted for these parameters. Values specified are the default values.

All parameters on this page are ignored by the target.

Control Parameters - Page x'0A'

Figure 29. Mode Page x'0A'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page Code=x'0A'					
1	Page Length=10 (x'0A')							
2	0							
3	Queue algorithm modifier				0		QErr	DQue=0
4	0	RAC=0	ByprtM=1	BybthS=1	0	RAERP=0	UAAERP=0	EAERP=0
5	0							
6-7	Ready AER holdoff period=0							
8-9	Busy timeout period=0							
10-11	0							

Queue Algorithm Modifier Setting this parameter to 0 restricts the targets freedom to reorder Simple Queue tagged commands. Overlapping Write operations from the same initiator update the media in the same order which the commands are received.

Note: Overlapping Write operations are write operations that update some of the same blocks on the logical unit.

Note: Setting this parameter to 0 is of no value when an interconnect technology is used that does not guarantee in-order transmission of commands.

Note: This parameter has no effect on Read operations.

Setting this parameter to 1 allows the target to reorder Simple Queue tagged commands in any way. This is the default setting.

QErr -queue error management Setting QErr=1 specifies that some or all outstanding commands for a logical unit be aborted when a contingent allegiance condition is cleared for that logical unit. This is the default setting. The QEMC parameter determines which commands are aborted. See "Vendor Unique Parameters - Page x'00'" on page 42 for a description of the QEMC parameter.

Setting QErr=0 specifies that outstanding commands for a logical unit be resumed when a contingent allegiance is cleared for that logical unit.

DQue -disable queuing This parameter must be set to 0. The target accepts commands with Queue tags.

RAC -report a check This parameter must be set to 0. When appropriate, the target gives Busy status.

ByprtM (by port message)

BybthS (by both reset) These parameters must be set to 1. A Target Reset message or a hard reset applies only to the port over which it is transacted. Ports which do not share any logical units with the port that receives a Reset is unaffected.

The Target Reset function causes the LUN Reset function to be executed for all logical units that are surfaced on the port. All outstanding commands for those logical units are aborted. Reset Unit Attention conditions are generated by the affected logical units for all initiators on the ports that surface these logical units.

Renegotiation of physical interface characteristics such as transfer speed are only required for the port which received the Reset.

RAERP (ready AER permission)

UAAERP (unit attention AER permission)

EAERP (error AER permission) These parameters must be set to 0. The target does not support asynchronous event reporting.

──────── Fibre channel Only ────────

Fibre Channel Logical Unit Control - Page x'18'

Figure 30. Mode Page x'18'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page Code=x'18'					
1	Page Length=6							
2	0							
3	0							EPDC=0
4-7	0							

This mode page can only be read or modified by fibre channel initiators.

EPDC   Enable Precise Delivery Checking (EPDC) is not supported by ESS.

Fibre Channel Port Control page x'19'

Figure 31. Mode Page x'19'

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0		Page Code=x'19'					
1	Page Length=6							
2	0							
3	0	PLPB=0	DDIS=0	DLM=0	DSA=0	ALWI=0	DTIPE=0	DTOLI=0
4-7	0							

This mode page can only be read or modified by fibre channel initiators and only on LUN zero.

PLPB   A Prevent Loop Port Bypass (PLPB) bit of 1 indicates that a target attached to an FC-AL loop shall ignore any Loop Port Bypass (LPB) and Loop Port Enable (LPE) primitive sequences. The loop port shall always remain participating. When PLPB bit is 0, the target allows the LPB and LPE primitive sequences to control the port bypass circuit and participation on the loop as specified by FC-AL. This bit is ignored by ports configured as point-to-point.

DDIS   A Disable Discovery (DDIS) bit of 1 indicates that a target attached to an FC-AL loop shall not require receipt of Address or Port Discovery (ADISC or PDISC ELSs) following loop initialization. The logical units shall resume processing tasks on completion of loop initialization. When the DDIS bit is 0, the target shall wait to complete target discovery as defined by FC-PLDA and FC-FLA before allowing processing of tasks to resume. This bit is ignored by ports configured as point-to-point.

DLM   A Disable Loop Master (DLM) bit of 1 indicates that a target attached to an FC-AL loop shall not participate in loop master arbitration and shall not become loop master. The target shall only repeat LISM frames it receives. When the DLM bit is 0, the target can participate in loop master arbitration in the normal manner and can become loop master during the loop initialization process. This bit is ignored by ports configured as point-to-point.

DSA   A Disable Soft Address (DSA) bit of 1 indicates that a target attached to an FC-AL loop shall not select a soft address if there is a conflict for the hard address selection during loop initialization. If there is a conflict, the target shall enter the nonparticipating state. If the target detects loop initialization while in the nonparticipating state, the target again attempts to get its hard address. When the DSA bit is 0, the target follows the normal initialization procedure, including the possibility of obtaining a soft address during the loop initialization process. This bit is ignored by ports configured as point-to-point.

ALWI   An Allow Login Without Loop Initialization (ALWLI) bit of 1 indicates that a target attached to an FC-AL loop shall use the hard address, enter the monitoring state in participating mode, and accept logins without using the loop initialization procedure. When the ALWLI bit is 0, the target shall perform the normal loop initialization procedure before entering the

monitoring mode and accepting a login ELS. Targets attached to an N_Port or to an F_Port shall ignore this bit.

DTIPE   A Disable Target Initiated Port Enable (DTIPE) bit of 1 indicates that a target attached to an FC-AL loop shall wait for an initiator to send the LPE primitive sequence before inserting itself into a loop. The target shall wait in a nonparticipating state with the Port Bypass circuit, if any, set to bypass the target. The target uses the hard address to determine whether LPE primitive sequences are addressed to it. An LPE primitive sequence addressed to the broadcast address shall also cause the target to insert itself into the loop. When the DTIPE bit is 0, the target shall enable itself onto the loop in a vendor-specific manner. This bit is ignored by ports configured as point-to-point.

DTOLI   A Disable Target Originated Loop Initialization (DTOLI) bit of 1 indicates that a target attached by an FC-AL loop shall not generate an Initialization LIP following insertion into the loop. The target shall respond to an Initializing LIP when it is received. When DTOLI bit is 0, the target attached by an FC-AL loop shall generate the Initializing LIP after it enables a port into a loop. If the target is attached to an FC-AL loop and detects loop failure at its input, it shall follow the error initialization process defined by FC-AL-2. This bit is ignored by ports configured as point-to-point.

The DTIPE and PLB bits shall not both be set to 1 at the same time. When an invalid bit combination is sent by the application client, the device server shall return Check Condition status and the sense key shall be set to illegal request with the additional sense code set to `Invalid Field in the Parameter List`.

—————————————— End of Fibre channel Only ——————————————

Mode Sense

Figure 32. Mode Sense

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'1A'							
1	Reserved				DBD	Reserved		
2	PC		Page code					
3	Reserved							
4	Allocation Length							
5	0		Reserved			NACA	0	

Figure 33. Mode Sense (10 byte)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'5A'							
1	Reserved				DBD	Reserved		
2	PC		Page code					
3-6	Reserved							
7-8	Allocation Length							
9	0		Reserved			NACA	0	

The Mode Sense command allows an initiator to retrieve logical unit parameters. The initiator can retrieve the current values, the default values, the saved values, or a mask indicating which parameters are changeable.

There is one set of parameters for each logical unit. The parameters are shared for all initiators on all ports that access the logical unit.

DBD   Setting DBD=0 causes the target to include one block descriptor in the Mode Sense data (following the header). See Figure 22 on page 42 for a description of the block descriptor.

Setting DBD=1 causes the target to return Mode Sense data which does not include a block descriptor.

PC -page control This field determines which values are returned.

0   The current values of the parameters are returned. These are the values for the parameters that the target is currently using. The current values can be different from the saved values if a Mode Select command with SP=0 has been executed since the logical unit was last reset (or powered on).

1   A mask is returned that indicates which parameters are changeable. Fields in the mode pages for parameters that can be changed with the Mode Select command are set to all 1's. Fields for unchangeable parameters will be set to all 0's.

Parameters in "Mode Pages" on page 42 that list only one valid value or that limit the value to be the same as that returned by Mode Sense are considered unchangeable. Also, fields in mode pages which are not named parameters are considered unchangeable.

2   The default values of the parameters are returned. These are the values for the parameters that the target uses before any parameters are changed by an initiator.

3   The saved values of the parameters are returned. These are the values that were received from an initiator which did a Mode Select command with SP=1. The saved values for all mode parameters are initially equal to their default values. After a logical unit is reset (or after power on), the current values are set to the saved values.

Page Code This field specifies which Mode Page is to be returned.

Page Code=3Fh specifies that all Mode Pages are to be returned. The pages are returned in ascending order (by page code) except that Page 00h is last.

If an unsupported page code is specified, Check Condition status is returned. The sense data indicates `Illegal Request-Invalid Field in CDB`.

The mode parameters are returned using Data In. The data begins with a header as defined in Figure 20 on page 41 "Mode Header". The header is followed by a block descriptor if DBD=0. The mode pages follow.

The Mode Sense command supports the same set of pages as does the Mode Select command. See 5.6.1 "Mode Pages" on page 34 for a list of supported pages and their definition. The pages returned by the Mode Sense command are formatted identically to those pages, with one difference. All pages returned by the Mode Sense command have the PS (parameters saveable) bit set to 1. See Figure 34: "Mode Sense Page Format".

Figure 34. Mode Sense Page Format

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PS=1	0	Page code					
1	Page Length							
2-n	Parameters							

Persistent Reserve In

Figure 35. Persistent Reserve In

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'5E'							
1	Reserved			Service Action				
2-6	Reserved							
7-8	Allocation Length							
9	0		Reserved			NACA	0	

The Persistent Reserve In command is used to obtain information about persistent reservations and reservation keys that are active. This command is used in conjunction with the Persistent Reserve Out command (see "Persistent Reserve Out" on page 54).

See "Persistent reserve" on page 23 for more information on how Persistent Reserve works.

Service Action Actions requested by Persistent Reserve In. Defined as follows:

> **Value Definition**
> **00h**   Read Keys - Reads all registered reservation keys
> **01h**   Read Reservation - Reads the current persistent reservation
> **02h-1Fh** Reserved

Allocation Length The actual length of the Persistent Reserve In parameter data is available in a parameter data field. The Allocation Length field in the CDB indicates how much space has been reserved for the returned parameter list. If the length is not sufficient to contain the entire parameter list, the first portion of the list is returned. This is not considered an error. If the remainder of the list is required, the application client sends a new Persistent Reserve In command with a Allocation Length field large enough to contain the entire list.

Parameter data for read keys

Figure 36. Persistent Reserve In parameter data for read keys

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0-3	Generation							
4-7	Additional Length (n-7)							
8-n	List of eight byte reservation keys							

Generation The Generation value is a 32-bit counter kept by volume that is incremented every time a Persistent Reserve Out command requests a Register, Clear, Preempt, or Preempt and Abort service action. The counter is not incremented by a Persistent Reserve In command, by a Persistent Reserve Out command that performs a Reserve or Release service action, or by a Persistent Reserve Out command that is not performed due to an error or reservation conflict. Regardless of the APTPL value, the Generation value is set to 0 as part of the power on reset process.

Additional Length The Additional Length field contains a count of the number of bytes in the reservation key list. If the Allocation Length specified by the Persistent Reserve In command is not sufficient to contain the entire parameter list, then only the bytes from 0 to the maximum allowed allocation length is sent to the initiator. The remaining bytes are truncated, although the Additional Length field still contains the actual number of bytes in the reservation key list without consideration of any truncation resulting from an insufficient allocation length. This is not considered an error.

Reservation Key List The reservation key list contains all the 8-byte reservation keys registered with the storage facility and LUN through a Register service action.

Parameter data for read reservation

Figure 37. Persistent Reserve In parameter data for read reservation

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0-3	Generation							
4-7	Additional Length (0 or 16)							
8-15	Reservation Key							
16-20	0							
21	0				Type			
22-23	0							

Generation See definition from Figure 36.

Additional Length The Additional Length field contains a 0 if the LUN is not reserved or 16 if it is reserved. If the allocation length specified by the Persistent Reserve In command is not sufficient to contain the entire parameter list, then only the bytes from 0 to the maximum allowed allocation length is sent to the initiator.

Reservation Key The eight byte reservation key that holds the reserve.

Type The type of reservation held

Value Definition

1h Write Exclusive

Reads Shared: Any initiator can execute commands that perform transfers from the storage medium or cache of the logical unit to the initiator.

Writes Exclusive: Any command from any initiator other than the initiator holding the persistent reservation that performs a transfer from the initiator to the storage medium or cache of the logical unit results in a reservation conflict.

3h Exclusive Access

Reads Exclusive: Any command from any initiator other than the initiator holding the persistent reservation that performs a transfer from the storage medium or cache of the logical unit to the initiator results in a reservation conflict.

Writes Exclusive: Any command from any initiator other than the initiator holding the persistent reservation registered with the key holding the persistent reservation that performs a transfer from the initiator to the storage medium or cache of the logical unit results in a reservation conflict.

5h Write Exclusive, Registrants Only

Reads Shared: Any initiator can execute commands that perform transfers from the storage medium or cache of the logical unit to the initiator.

Writes Exclusive: Any command from an initiator that has not previously performed a Register service action with the LUN that performs a transfer to the storage medium or cache of the logical unit, results in a reservation conflict.

6h Exclusive Access, Registrants Only

Reads Exclusive: Any command from an initiator that has not previously performed a Register service action with the LUN that performs a transfer from the storage medium or cache of the logical unit, results in a reservation conflict.

Writes Exclusive: Any command from an initiator that has not previously performed a Register service action with the LUN that performs a transfer to the storage medium or cache of the logical unit, results in a reservation conflict.

Persistent Reserve Out

See "Persistent reserve" on page 23 for more information on how Persistent Reserve works.

Figure 38. Persistent Reserve Out

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'5F'							
1	Reserved			Service Action				
2	0				Type			
3-6	Reserved							
7-8	Parameter List Length=18h							
9	0		Reserved			NACA	0	

Service Action Actions requested by Persistent Reserve Out. Defined as follows:

> **Value Definition**
>
> **00h** Register - Register a reservation key
>
> **01h** Reserve - Create a persistent reservation using a reservation key
>
> **02h** Release - Release a persistent reservation
>
> **03h** Clear - Clear all reservation keys and all persistent reservations
>
> **04h** Preempt - Preempt persistent reservations from another key
>
> **05h** Preempt and Abort - Preempt persistent reservations from another key and abort the task set for the initiators registered with that key
>
> **06h-1Fh** Reserved

Type See definition from Figure 37 on page 53.

The parameter list shall be 24 bytes in length and the parameter list length field shall contain 24 (18h). If the parameter list length is not 24, the command is terminated with Check Condition status. The sense key shall be set to Illegal Request and the additional sense data shall be set to Parameter List Length Error.

Figure 39. Persistent Reserve Out Parameter List

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0-7	Reservation Key							
8-15	Service Action Reservation Key							
16-19	0							
20	Reserved							APTPL
21	Reserved							
22-23	0							

Reservation Key The Reservation key field contains an 8-byte value provided by the initiator to the storage facility to identify the initiator that is the source of the Persistent Reserve Out command. The Reservation key field in a Persistent Reserve Out command must match the registered reservation key for the initiator from which the command was received, except for the Register service action for an unregistered initiator which shall have a reservation key value of 0. If a Persistent Reserve Out command specifies a Reservation key field other than the reservation key registered for the initiator, Reservation Conflict status is returned. The reservation key of the initiator is valid for all service action values.

Service Action Reservation Key The Service Action Reservation key field contains information needed for the Register, Preempt, and Preempt and Abort service actions. For the Register service action, the Service Action Reservation key field contains the new reservation key to be registered. For the Preempt and Preempt and Abort service actions, the Service Action Reservation key field contains the reservation key of the persistent reservations that are being preempted. The Service Action Reservation key is ignored for all service actions except those described in this paragraph.

APTPL The Activate Persist Through Power Loss (APTPL) bit is valid only for the Register service action. In all other cases, the APTPL is ignored

APTPL is kept by the storage facility on a LUN basis. When APTPL is in effect on a LUN, persistent reservations and reservation keys are retained across a power loss or machine IML. When APTPL is not in effect on a LUN, persistent reservations and reservation keys are cleared by a power loss or machine IML.

> **Note:** Some hardware failures can cause this information to be lost even if APTPL is set. In these cases, Unit Attention (power on) is primed for all initiators.

Register service action

The registration request is processed regardless of any active persistent reservations. The reservation key is registered without generating any persistent reservation. After the registration request has been processed, Persistent Reserve Out commands from the registered initiator are allowed to run.

Once registered, the reservation key is retained until either the key is changed by that initiator or until the initiator registration is removed by one of the following actions:

- Powering down or IMLing the storage facility, unless APTPL is on

- Performing a Clear service action

- Performing a Preempt service action

- Performing a Preempt and Abort service action

- Performing a Register service action from the same initiator, with the value of the Service action Reservation key field set to 0.

- On fibre channel, the initiator logs out

When a reservation key has been removed, no information is reported for the initiator that unregistered initiator in the Read Keys service action. If removing a reservation key causes a persistent reserve of either Write Exclusive, Registrants Only or Exclusive Access, Registrants Only to be released; then a Unit Attention (Reservation Released) is primed for all initiators registered with that LUN.

Any Persistent Reserve Out command service action received by an unregistered initiator, other than the Register service action, results in the command being rejected with a Reservation Conflict status.

It is not an error for an initiator that is registered to reregister with the same reservation key or a new reservation key.

If the register service action is registering a new initiator on this LUN and there are not enough resources to accept another registration, then the command is rejected (Insufficient Registration Resources).

If the initiator has indicated on any LUN via Set System Characteristics order that it supports host messages then the initiator is only allowed to use one reservation key across all LUNS.

Reserve service action

This service action is used to reserve a LUN with a persistent reserve of the Type specified in the Type field of the Persistent Reserve Out parameters.

This request must come from an initiator that is registered with the LUN and the LUN must not currently be reserved. If these conditions are not met, the request is rejected with a status of Reservation Conflict.

Release service action

This service action is used to reset a persistent reserve on a LUN.

This request must come from the initiator that is holding the reserve. If this condition is not met, the request is ignored and Good status is returned.

If a persistent reserve is released that is either of the type write exclusive, registrants only or exclusive access, registrants only; then a Unit Attention (Reservation Released) is primed for all other registered initiators on that LUN.

Preempt service action

This service action is used to remove a registration or steal a persistent reservation.

Preempting causes the following to happen:

- If the LUN is reserved to an initiator that is registered with the Preempted Reservation key then the reservation type is set to the type specified by the Preempt service action without regard to its previous setting.
- With the exception of the initiator that issued the preempt the registration is reset for all initiators registered with the Reservation key that was preempted.
- A Unit Attention (Registration Preempted) is primed for the initiators whose registration was reset as a result of the preempt.
- If the Reserve that was preempted was either of the type write exclusive, registrants only or exclusive access, registrants only and the new reservation type is not the same; then a Unit Attention (Reservation Released) is primed for all of the initiators still registered on that LUN, except for the initiator that issued the preempt.

Preempt and Abort service action

This service action is used to remove a registration or steal a persistent reservation and to abort tasks on the LUN that originated from the initiators that are registered with the preempted key.

Preempting causes the following to happen:

- All of the actions specified in "Preempt service action."
- If the LUN is in a Contingent Allegiance or an Auto Contingent Allegiance with an initiator that is registered with the LUN with the Preempted reservation key then the CA/ACA is reset.
- All tasks queued on the LUN by initiators that are registered with the reservation key of the preempted reserve are aborted as if there were an abort task for each task from those initiators.

Clear service action

This service action is used to reset any persistent reserve and to clear all reservation key registrations for this LUN.

Once cleared a Unit Attention (Reservation Preempted) is primed for all initiators, except the one that issued the clear, that were registered on that LUN.

Read

Figure 40. Read (6 Byte command)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'08'							
1	Reserved			Logical Block Address				
2-3	Logical Block Address							
4	Transfer Length							
5	0		Reserved			NACA	0	

Figure 41. Read (10 Byte command)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'28'							
1	Reserved			DPO	FUA	Reserved		0
2-5	Logical Block Address							
6	Reserved							
7-8	Transfer Length							
9	0		Reserved			NACA	0	

The Read command requests that the target transfer data to the initiator. The most recent data written to the addressed blocks is returned.

DPO   Disable Page Out. When set to 1, this indicates that the blocks accessed should be assigned the lowest priority in cache.

FUA   Force Unit Access. When set to 1, this indicates that the blocks should be read directly from the disk.

Logical Block Address This field specifies the block at which the read operation is to begin. The blocks are read sequentially.

Transfer Length This field specifies the number of contiguous blocks to be transferred.

> For the 6-byte Read command (Operation Code=08h), Transfer Length=0 specifies that 256 blocks are to be transferred.

> For the 10-byte Read command (Operation Code=28h), Transfer Length=0 specifies that no data is to be transferred. This is not an error.

Read Buffer

Figure 42. Read Buffer

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'3C'							
1	Reserved					Mode		
2	Buffer ID							
3-5	Buffer offset							
6-8	Allocation Length							
9	0		Reserved			NACA	0	

The Read Buffer command is used in conjunction with the Write Buffer command as a diagnostic for testing the integrity of the service delivery subsystem. The Read Buffer command returns the contents of the targets buffer. This buffer is used for several commands. Therefore, the Read Buffer command should be issued immediately following a Write Buffer command (without other commands between them) if the initiator desires to compare the written data to the read data.

There is one buffer per port. It may be modified or accessed by any command sent to any target or LUN surfaced on the port.

Mode This field must be set to 0. Only Combined Header and Data mode is supported.

Buffer ID This field must be set to 0. The target only has one buffer.

Buffer Offset This field must be set to 0. The data is read starting at the beginning of the buffer.

A 4-byte header is returned followed by data from the buffer. The number of bytes sent using Data Out is either the Allocation Length or 512 bytes (whichever is less). The header is shown in Figure 43 "Read Buffer Header".

Figure 43. Read Buffer Header

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0							
1-3	Buffer capacity=508							

Read Capacity

Figure 44. Read Capacity

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'25'							
1	Reserved							0
2-5	Logical Block Address							
6-7	Reserved							
8	Reserved							PMI
9	0		Reserved			NACA	0	

> **Logical Block Address** This field is used in conjunction with the PMI bit. When PMI = 0 this shall be zero or a Check Condition is indicated with a sense key of Illegal Request and additional sense code of Illegal field in the CDB.
>
> When PMI = 1, the host is attempting to discover which block address starts a region where a substantial delay in data transfer occurs. The host is interested in the next break where this occurs between the logical block address and the last logical block address. Since our product does not exhibit this behavior, no special action is taken other than not checking the value of the logical block address.
>
> **PMI** Partial Medium Indicator. See Logical Block Address for how this is used.

The Read Capacity command returns information about the capacity of the logical unit using Data In. The data is formatted as shown in Figure 45, "Read Capacity Data".

Figure 45. Read Capacity Data

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0-3	Last Logical Block Address							
4-7	Block Length							

> **Last Logical Block Address** The logical block address of the last logical block on the logical unit.
>
> **Block Length** Length in bytes of each logical block on the logical unit.

Reassign Blocks

Figure 46. Reassign Blocks

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'07'							
1-4	Reserved							
5	0		Reserved			NACA	0	

> The Reassign Blocks command corrects media errors such that subsequent attempts to read the sector does not receive Check Condition status (with Sense

Key=Media Error). The data returned by such a read command would not be valid. The data for a reassigned sector is lost.

For ESS the effect of a reassign is to rewrite the affected sectors with zeros.

The Reassign Blocks command only supports reassigning one block per command.

The initiator specifies which logical block is to be reassigned in a data structure that is sent using Data Out. The format of this data structure is shown in Figure 47, "Reassign Blocks Defect List".

Figure 47. Reassign Blocks Defect List

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0-1	0							
2-3	Defect list length=4							
4-7	Defect logical block address							

Report LUNs

Figure 48. Report LUNs

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'A0'							
1-5	Reserved							
6-9	Allocation Length							
10	Reserved							
11	0		Reserved			NACA	0	

This command returns the list of LUNs that are configured to this target or initiator.

The Report LUNs command neither presents Unit Attention status if it is pending, nor does it reset that status.

See "Controlling initiator access to FB volumes" on page 16 for how LUNs are configured to targets and initiators.

Allocation Length defines the maximum number of bytes that can be sent by this command. Allocation length must be at least 16. If the allocation length is less than 16 bytes, the command is terminated with Check Condition status. The sense key is set to Illegal Request and the additional sense data is set to Invalid Field in CDB.

The format of the data returned is defined by the following table.

Figure 49. Report LUNS Data

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0-3	LUN list length							
4-7	0							
8-n	List of 8 byte LUN IDs							

LUN list length Number of bytes available in the list of 8 byte LUN IDs. This number is not changed if the allocation length is too small to allow the transfer of the entire LUN list.

LUN ID ESS uses only one level of addressing and only uses logical addressing. See the following table for the format of the LUN ID.

Figure 50. LUN_ID

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
For fibre channel hosts, configured as using Report LUNs:								
0	Addr_Method '01'b (Virtual Device)		LUN					
1	LUN							
2-7	0							
For fibre channel hosts, configured not as using Report LUNs:								
0	Addr_Method '00'b (Peripheral Device)		BUS = 0					
1	LUN							
2-7	0							
For parallel SCSI hosts:								
0	Addr_Method '00'b (Peripheral Device)		BUS = 0					
1	0		LUN					
2-7	0							

Request Sense

Figure 51. Request Sense

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'03'							
1-3	Reserved							
4	Allocation length							
5	0		Reserved			NACA	0	

The Request Sense command requests that the target transfer sense data to the initiator.

The sense data returned depends on the conditions in the LUN and the storage control unit as follows:

1. If there is valid sense data for the initiator because of a Check Condition status returned on the prior command, then that sense data is returned.

2. If a Unit Attention is active (for the initiator and LUN combination), the sense data for the Unit Attention is returned and the Unit Attention condition is cleared.

   Note: The Unit Attention condition is preserved for the initiator and LUN combination until the sense data reporting it has been sent to the initiator.

3. Otherwise, the sense buffer associated with the initiator is checked to see if the last operation with that initiator was deemed to have a protocol error, causing the target to abort and go Bus Free.

   Note: This buffer is reset every time any target in the storage control presents status to the initiator.

4. Otherwise there is no valid sense data, a sense key of 0 is returned.

The sense data is transferred using Data In. See Chapter 7, "Status and sense data" on page 75 for a definition of the sense data contents.

If a Request Sense command is issued with an invalid LUN, sense data is returned indicating `Illegal Request - Logical Unit Not Supported.`

See "Auto contingent allegiance condition" on page 20 for a description of the events that follow a Check Condition.

Should a Check Condition occur during the execution of the Request Sense command, then the sense data that was being presented is lost and the reason for the Check Condition on the Request Sense command is presented on the next Request Sense command. Unit Attention condition is not lost in this manner, instead the Unit Attention is represented after the sense data for the Check Condition has been presented.

Release

Figure 52. Release (6 byte)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'17'							
1	Reserved			0				
2	Ignored							
3-4	Reserved							
5	0		Reserved			NACA	0	

Figure 53. Release (10 byte)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'57'							
1	Reserved			3rdPty	Reserved		LongID	0
2	Ignored							
3	3rd Party ID							
4-6	Reserved							
7-8	Parameter List Length							
9	0		Reserved			NACA	0	

The Release command is used to release a previously reserved logical unit.

It is not an error for an initiator to attempt to release a reservation that is not currently active. In this case, the target returns Good status without altering any reservations.

3rdPty   3rdPty=0 specifies that the Release command corresponds to a reservation held by the initiator issuing the command. The 3rd Party ID field is ignored.

3rdPty=1 specifies that the Release command corresponds to a reservation held by the initiator specified by the 3rd Party ID field.

If 3rdPty=1 and the initiator is attached using a fibre channel port then LongID must also be set to 1. Otherwise the command is terminated with Check Condition status with sense data set to `Illegal Request-Invalid Field in CDB`.

Long ID   LongID=0 specifies that the 3rd Party ID field in the CDB is valid. If 3rdPty=1, then the release occurs for the initiator specified in 3rd Party ID. The Parameter List Length must be 0.

LongID=1 specifies that the 3rd Party ID field in the CDB is to be ignored. The target receives the third party ID using Data Out. The format of this data is specified in Figure 56 on page 65 "Reserve/Release Parameter List". The Parameter List Length must be 8. If 3rdPty=1, then the release occurs for the initiator specified in the parameter list.

Only the initiator that issued the reserve command for a logical unit can release the logical unit. This initiator can also release the logical unit by issuing another Reserve command. When the new reservation is granted, the previous one is implicitly released.

Reserve

Figure 54. Reserve (6 byte)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'16'							
1	Reserved			0				
2	Ignored							
3-4	Reserved							
5	0		Reserved			NACA	0	

Figure 55. Reserve (10 byte)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'56'							
1	Reserved			3rdPty	Reserved		LongID	0
2	Ignored							
3	3rd Party ID							
4-6	Reserved							
7-8	Parameter List Length							
9	0		Reserved			NACA	0	

The Reserve command is used to reserve an entire logical unit for an initiator.

3rdPty 3rdPty=0 specifies that the logical unit be reserved for the initiator issuing the command. The 3rd Party ID field and the LongID flag are ignored.

3rdPty=1 specifies that the logical unit be reserved for the initiator specified by the 3rd Party ID field. If the initiator specified by 3rdParty ID is the initiator that issued the Reserve command, the reservation is handled as if 3rdPty were 0.

If 3rdPty=1 and the initiator is attached using a fibre channel port then LongID must also be set to 1. Otherwise the command is terminated with Check Condition status with sense data set to `Illegal Request-Invalid Field in CDB`.

A 3rd party reservation may only be released with a 3rd party release.

Long ID LongID=0 specifies that the 3rd Party ID field in the CDB is valid. If 3rdPty=1, then the reservation is made for the initiator specified in 3rd Party ID. The Parameter List Length must be 0.

LongID=1 specifies that the 3rd Party ID field in the CDB is to be ignored. The target receives the 3rd party ID using Data Out. The format of this data is specified in Figure 56 "Reserve/Release Parameter List". The Parameter List Length must be 8. If 3rdPty=1, then the reservation is made for the initiator specified in the parameter list.

Figure 56. Reserve/Release Parameter List

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Format for a parallel SCSI initiator:								
0-6	0							
7	0			3rd Party ID				
Format for a fibre channel initiator:								
0-7	World Wide Port Name							

The reservation exists until one of these events occurs:

- The initiator that issued the reserve command issues another Reserve command for the same logical unit.

- The initiator that issued the Reserve command issues a Release command for the logical unit.

- The logical unit is reset by a LUN Reset message, a Target Reset message, or a hard reset.

- A power off and on cycle occurs.

Reservation queuing is not supported. If a logical unit is reserved and a different initiator issues a reserve command, Reservation Conflict status is returned.

Send Diagnostic

Figure 57. Send Diagnostic

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'1D'							
1	Reserved			0	Reserved	SelfTest	0	
2	Reserved							
3-4	0							
5	0	Reserved				NACA	0	

The Send Diagnostic command requests the target to run self-diagnostic tests. The target tests its ability to read and write the media for the logical unit. Good status is returned if the tests pass.

Check Condition status is returned if one or more tests fail. The sense data indicates `Hardware Error or Not Ready-Media Format Corrupted, Reassign Failed.`

Self Test This field must be set to 1. A self test is executed.

Skip Read

Figure 58. Skip Read

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =X'E8'							
1	Reserved			0				
2-5	Logical block address							
6	Mask Length							
7-8	Transfer length							
9	0	Reserved				NACA	0	Link=1

The Skip Read command requests that the target execute a Read command for non contiguous logical blocks.

Logical Block Address This field specifies the block at which the read operation is to begin.

Mask Length This field specifies the length in bytes of the skip mask.

Mask length=0 specifies a mask length of 256.

Transfer Length This field specifies the number of blocks to be transferred.

Transfer Length=0 specifies that no data is to be transferred. This is not an error.

The maximum transfer length is 256 blocks. If a larger number is specified, then Check Condition status is returned. The sense data is set to `Illegal Request-Invalid Field in CDB.`

This command accepts a skip mask using Data Out. The number of bytes accepted using Data Out is specified by Mask Length.

The target sends an intermediate Good status and a Linked Command Complete message, then it accepts a second command.

The target sends an I5 IU (Command Response (linked)) then it accepts a second command in a T3 IU (Command Request (linked)).

The second command must be a Read command. It may be either a 6- or 10-byte Read command. For parallel SCSI, the target does not disconnect before receiving the second command.

Each bit in the skip mask corresponds to a logical block. The first bit of the skip mask corresponds to the logical block address specified in the CDBs. Each subsequent bit in the skip mask corresponds to the next sequential logical block address. The target transfers the logical blocks which correspond to 1's in the skip mask.

The number of bits that are set to 1 in the skip mask must be equal to the Transfer Lengths specified in both the Skip Read CDB and the Read CDB. If these three numbers are not equal, Check Condition status is returned. The sense data is set to `Illegal Request-Invalid Field in CDB`.

If the second command is not a Read command or if the LBA of the Read command does not match the LBA of the Skip Read command, then Check Condition status is returned. The sense data is set to `Illegal Request-Invalid Field in CDB`.

Skip Write

Figure 59. Skip Write

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =X'EA'							
1	Reserved			0				
2-5	Logical block address							
6	Mask Length							
7-8	Transfer length							
9	0		Reserved			NACA	0	Link=1

The Skip Write command requests that the target execute a write command for noncontiguous logical blocks.

Logical Block Address This field specifies the block at which the write operation is to begin.

Mask Length This field specifies the length in bytes of the skip mask.

> Mask length=0 specifies a mask length of 256.

Transfer Length This field specifies the number of blocks to be transferred.

> Transfer Length=0 specifies that no data is to be transferred. This is not an error.

> The maximum transfer length is 256 blocks. If a larger number is specified, then Check Condition status is returned. The sense data is set to `Illegal Request-Invalid Field in CDB`.

This command accepts a skip mask using Data Out. The number of bytes accepted using Data Out is specified by Mask Length.

```
┌─────────────────────────── Parallel SCSI attached ───────────────────────────┐
```

The target sends an intermediate Good status and a Linked Command Complete message, then it accepts a second command.

```
└─────────────────────────── End of Parallel SCSI attached ─────────────────────┘
```

```
┌─────────────────────────────── FCP attached ─────────────────────────────────┐
```

The target sends an I5 IU (Command Response (linked)) then it accepts a second command in a T4 IU (Command Request (linked)).

```
└─────────────────────────────── End of FCP attached ──────────────────────────┘
```

The second command must be a Write command. It may be either a 6- or 10-byte Write command. For parallel SCSI, the target does not disconnect before receiving the second command.

Each bit in the skip mask corresponds to a logical block. The first bit of the skip mask corresponds to the logical block address specified in the CDBs. Each subsequent bit in the skip mask corresponds to the next sequential logical block address. The target transfers the logical blocks which correspond to 1's in the skip mask.

The number of bits that are set to 1 in the skip mask must be equal to the Transfer Lengths specified in both the Skip Write CDB and the Write CDB. If these three numbers are not equal, Check Condition status is returned. The sense data is set to `Illegal Request-Invalid Field in CDB`..

If the second command is not a Write command or if the LBA of the Write command does not match the LBA of the Skip Write command, then Check Condition status is returned. The sense data is set to `Illegal Request-Invalid Field in CDB`.

Note: As a result of AS/400 requirement this command does not disconnect until after the skip mask has been transferred.

Start/Stop Unit

Figure 60. Start/Stop Unit

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'1B'							
1	Reserved							Immed
2-3	Reserved							
4	Reserved						0	Start
5	0		Reserved			NACA	0	

The Start/Stop Unit command requests that the target enable and disable the logical unit for all media access operations.

Immed Immed=0 specifies that status is to be returned at the end of the operation.

Immed=1 specifies that status is to be returned immediately for the command. Good status is returned unless the target has an error or Unit Attention condition to report.

Start Start=0 specifies that the logical unit be stopped. ESS accepts and ignores the request to stop the logical unit.

Start=1 specifies that the logical unit be started. After the logical unit is started successfully, it accepts and runs media access commands.

Note: The logical units autostart. A Start Unit command is not required after power on.

It is not an error for an initiator to start a logical unit that is already started.

It is not an error for an initiator to stop a logical unit that is already stopped.

Synchronize Cache

Figure 61. Synchronize Cache

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'35'							
1	Reserved						Immed	0
2-5	Logical Block Address							
6	Reserved							
7-8	Number of blocks							
9	0		Reserved			NACA	0	

The Synchronize Cache command requests that the target force modified data from cache to the disk. ESS treats this command as a NOP. There is no checking of the values placed in the LBA or Number of Blocks fields.

Immed When the initiator sets this bit to 1, it is asking the target to return operation complete as soon as the command has been verified.

When the initiator sets this bit to 0, it is asking the target to return operation complete only when the operation has completed.

Logical Block Address This field specifies the block at which the operation is to begin.

Number of Blocks This field specifies the number of contiguous blocks to be synced.

Test Unit Ready

Figure 62. Test Unit Ready

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'00'							
1-4	Reserved							
5	0		Reserved			NACA	0	

The Test Unit Ready command allows the initiator to check if the logical unit is ready.

If the logical unit can accept and execute any media access command, then it returns Good status to the Test Unit Ready command. Otherwise, the Test Unit Ready command returns Check Condition status. The sense data indicates one of these conditions:

- Not Ready - Initializing command required
- Not Ready - Logical unit is in the process of becoming ready
- Not Ready - Logical unit not ready, format in progress
- Hardware Error

Verify

Figure 63. Verify

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'2F'							
1	Reserved			0	Reserved		0	
2-5	Logical block address							
6	Reserved							
7-8	Verification length							
9	0		Reserved			NACA	0	

The Verify command requests that the target access the addressed logical blocks. It returns Good status if the blocks are accessed without media errors.

Logical Block Address This field specifies the block at which the verification is to begin.

Verification Length This field specifies the number of contiguous blocks to be verified. Verification Length=0 specifies that no logical blocks are to be verified. This is not an error.

Write, Write and Verify

Figure 64. Write (6 Byte command)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'0A'							
1	Reserved			Logical Block Address				
2-3	Logical Block Address							
4	Transfer Length							
5	0		Reserved			NACA	0	

Figure 65. Write (10 Byte command)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'2A'							
1	Reserved			DPO	FUA	Reserved		0
2-5	Logical Block Address							
6	Reserved							
7-8	Transfer Length							
9	0		Reserved			NACA	0	

Figure 66. Write and Verify

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'2E'							
1	Reserved			0	Reserved		0	
2-5	Logical Block Address							
6	Reserved							
7-8	Transfer Length							
9	0		Reserved			NACA	0	

The Write command requests that the target receive data from the initiator. The target records the data on the media in the addressed logical blocks. The last data written to a logical block can be retrieved by the initiator with a Read operation.

If the command does not terminate successfully, then each addressed logical block contains either the data that was present before the write operation or the new data for that logical block that was transferred without error.

DPO   Disable Page Out. When set to 1, this indicates that the blocks accessed should be assigned the lowest priority in cache.

FUA   Force Unit Access. When set to 1, this indicates that the blocks should be written directly to the disk.

Logical Block Address This field specifies the block at which the write operation is to begin. The blocks are written sequentially.

Transfer Length This field specifies the number of contiguous blocks to be written.

For the 6-byte Write command (Operation Code=0Ah), Transfer Length=0 specifies that 256 blocks are to be written.

For the 10-byte Write command (Operation Code=2Ah), Transfer Length=0 specifies that no data is to be written. This is not an error.

The target does not implement any extra verification processes for the Write and Verify command. The processing is identical to that of the Write command.

Write Buffer

Figure 67. Write Buffer

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'3B'							
1	Reserved					Mode		
2	Buffer ID							
3-5	Buffer offset							
6-8	Parameter list length							
9	0		Reserved			NACA	0	

The Write Buffer command is used in conjunction with the Read Buffer command as a diagnostic for testing the integrity of the service delivery subsystem. The Write Buffer command writes data received from the initiator into the port's buffer. This buffer is used for several commands. Therefore, the Read Buffer command should be issued immediately following a Write Buffer command (without other commands between them) if the initiator desires to compare the written data to the read data.

There is one buffer per port. It may be modified or accessed by any command sent to any target/LUN surfaced on the port.

Mode   This field must be set to 0. Only Combined Header and Data mode is supported.

Buffer ID This field must be set to 0. The target only has one buffer.

Buffer Offset This field must be set to 0. The data is written starting at the beginning of the buffer.

It is not an error for the initiator to specify Parameter List Length=0. In this case, no data is written to the buffer.

The parameter list must begin with a 4-byte header which the target ignores. The data that is stored in the buffer begins with the fifth byte of the parameter list. If the data (excluding the header) would exceed the buffer size of the target, then Check Condition status is returned. The sense data indicates `Illegal Request - Parameter List Length Error.` The size of the target's buffer can be determined by issuing a Read Buffer command. The size of the buffer is returned in the 4-byte header.

Write Same

Figure 68. Write Same

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Operation code =x'41'							
1	Reserved					0		
2-5	Logical block address							
6	Reserved							
7-8	Number of blocks							
9	0		Reserved			NACA	0	

The Write Same command requests that the target receive one single block of data from the initiator. The target records this block of data on the media in each of the addressed logical blocks. The last data written to a logical block can be retrieved by the initiator with a Read command.

Logical Block Address This field specifies the block at which the write operation is to begin.

Number of Blocks This field specifies the number of contiguous blocks to be written. A value of 0 specifies that all blocks with LBAs greater or equal to the specified LBA be written.

Chapter 7. Status and sense data

These are the status codes returned by the target at the termination of a command.

00h Good status

The command completed successfully.

02h Check Condition status

An error has occurred. The sense data is valid. A Contingent Allegiance Condition exists for the initiator that issued the failed command.

08h Busy status

Another initiator has a Contingent Allegiance. The refused command had NACA=0.

18h Reservation Conflict status

The logical unit is reserved by another initiator.

28h Queue Full status

The target is unable to accept the command because its resources are allocated to other commands.

30h ACA Active status

Another initiator has a Contingent Allegiance. The refused command had NACA=1.

Sense data

Sense data is preserved by the logical unit for each command that completes with Check Condition status. The sense data is only guaranteed to be valid while a Contingent Allegiance Condition exists. The sense data can be valid at other times. If a Request Sense command is received when the sense data is not valid, then Sense Key=No Sense is returned. See "Request Sense" on page 62 for more information on how sense is offloaded.

There are several different internal conditions that can cause Check Condition status to be reported and sense data to be generated. It is possible for more than one of these conditions to be active at the same time. The order in which the target checks for each of these conditions determines their priority (highest priority first) as follows:

1. Reservation Conflict status
2. Busy or ACA Active status
3. Queue Full status
4. Check Condition status for incorrect initiator connection
5. Check Condition status for invalid logical unit number
6. Check Condition status for either initiator detected or message parity error
7. Check Condition status for Unit Attention condition
8. Check Condition status for Internal Error condition
9. Check Condition status for Format Failed condition
10. Check Condition status for Async Maintenance condition
11. Check Condition status for Active LUN condition

12. Check Condition status for invalid command opcode

13. Check Condition status for invalid command descriptor block

The format of the sense data is defined in Figure 69 "Sense Data Format".

Figure 69. Sense data format

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Valid	x'70'						
1	0							
2	0				Sense Key			
3-6	Information							
7	Additional Length=x'18'							
8-11	Command specific information							
12	Additional Sense Code							
13	Additional Sense Code Qualifier							
14	Field Replaceable Unit=0							
15	SKSV	Sense Key Specific Information						
16-17	Sense Key Specific Information							
18	Check Condition reason code, see Microcode DR3 for the definition of the values.							
19	Abort reason code, see Microcode DR3 for the definition of the values.							
20-21	URC, see Figure 72 on page 77 for values.							
22-23	Error Symptom code							
24-31	Other Interesting Engineering Data							

Figure 70. Sense Key Specific Information for Invalid Command

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
15	SKSV=1	C/D	0		BPV	Bit Pointer		
16-17	Field Pointer							

Figure 71. Sense Key Specific Information for Format In Progress

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
15	SKSV=1	0						
16-17	Progress Indication							

Valid Valid=0 indicates that the Information field is not valid. The Information field will is to 0.

Valid=1 indicates that the Information field is valid.

Information If the error was caused by a particular block, this field contains the logical block address of that block. For read or write operations, this is one of the logical blocks addressed by the command.

Command Specific Information If a Reassign Blocks command failed, this field contains the logical block address from the defect descriptor block that was not reassigned.

SKSV If Sense Key=Invalid Request because of a specification error in the Command Descriptor Block or in a parameter list, then SKSV = 1.

SKSV=0 indicates that the sense key specific information is invalid and is set to 0.

SKSV=1 indicates that the sense key specific information is valid. Its format is defined in Figure 70 on page 76, "sense key specific information for Invalid Command" and Figure 71 on page 76, "Sense Key Specific Information for Format In Progress".

C/D C/D=0 indicates that the error was in a parameter list sent using Data Out .

C/D=1 indicates that the error was in the Command Descriptor Block.

BPV BPV=0 indicates that the Bit Pointer field is invalid and is set to 0.

BPV=1 indicates that the Bit Pointer field is valid.

Bit Pointer This field indicates which bit of the byte designated by Field Pointer is in error. When a multi-bit field is in error, Bit Pointer indicates the most significant (leftmost) bit of the field.

Field Pointer This field indicates which byte of the Command Descriptor Block or parameter list is in error. Bytes are numbered starting with 0. When a multiple byte field is in error, Field Pointer indicates the most significant byte (lowest numbered) byte of the field.

Progress Indication This field describes the fraction of total logical unit capacity that has been formatted by an in progress format operation. The number in this field is the numerator of a fraction with 65536 as the denominator.

The Sense Key, Additional Sense Code, and Additional Sense Code Qualifier fields can be any of the combinations identified in Figure 72.

Figure 72 (Page 1 of 2). Sense key, code and qualifier combinations

Key	Code	Qualifier	Description	URC
00h	00h	00h	No additional sense information	0000h
02h	04h	01h	Logical unit is in the process of becoming ready	0000h
02h	04h	02h	Logical unit not ready, initialization command required	0000h
02h	04h	04h	Logical unit not ready, format in progress (see note)	0000h
02h	31h	00h	Media format corrupted, reassign failed	0000h
02h	31h	01h	Logical unit not ready, Format command failed	0000h
03h	0Ch	02h	Write error - auto reallocation failed	C0AAh
03h	11h	0Bh	Unrecovered read error, recommend reassignment	C0AAh
04h	44h	00h	Internal target failure	C904h
05h	1Ah	00h	Parameter list length error (see note)	0000h
05h	20h	00h	Invalid command operation code (see note)	0000h
05h	21h	00h	Logical Block Address out of range (see note)	0000h
05h	24h	00h	Invalid field in CDB (see note)	0000h
05h	25h	00h	Logical unit not supported	0000h

Figure 72 (Page 2 of 2). Sense key, code and qualifier combinations

Key	Code	Qualifier	Description	URC
05h	26h	00h	Invalid field in parameter list (see note)	0000h
05h	3Dh	00h	Invalid bits in identify message	0000h
05h	49h	00h	Invalid message error	0000h
06h	29h	00h	Power on, Reset, Target Reset, or LUN Reset occurred	0000h
06h	2Ah	01h	Mode parameters changed	0000h
06h	2Ah	03h	Reservation preempted	0000h
06h	2Ah	04h	Reservation released	0000h
06h	2Ah	05h	Registrations preempted	0000h
06h	2Fh	00h	Commands cleared by another initiator	0000h
0Bh	08h	02h	Logical unit communication parity error (LRC or sequence number error)	CFD8h
0Bh	1Bh	00h	Synchronous data transfer error	CFDFh
0Bh	2Ch	00h	Command sequence error	CFDFh
0Bh	43h	00h	Message error	CFDFh
0Bh	45h	00h	Select or Reselect failure	CFDFh
0Bh	47h	00h	SCSI parity error	CFDFh
0Bh	48h	00h	Initiator detected error message received	CFEFh
0Bh	49h	00h	Invalid message error	CFEFh
0Bh	4Eh	00h	Overlapped commands attempted	CFDFh
0Bh	55h	02h	Insufficient registration resources	0000h

Note: Sense Key Specific Data is provided.

Priorities on Unit Attention

When there is initative to offload more than one Unit Attention, the one with the lower priority is discarded. Unit Attentions have the following relative priority:

1. Power on/Reset occurred (highest)

2. One of the persistent reservation types

3. Commands cleared by another initiator

4. Mode page changed (lowest)

Chapter 8. Parallel SCSI

Messages

Attention Condition

An initiator causes an Attention Condition by asserting (and keeping asserted) the Attention signal before the selection phase starts.

An initiator causes an Attention Condition by asserting (and keeping asserted) the Attention signal before sending an Ack (in response to a Req from the target).

The target services the Attention Condition by going to the Message Out phase and accepting messages. The initiator continues to assert the Attention signal until it sees the Req for the last byte that it wants to send in the Message Out phase. The initiator must deassert the Attention signal before it sends the Ack for this last byte of the Message Out phase. The target can respond with appropriate messages in the Message In phase.

The target recognizes the Attention Condition at different times, depending on the current phase.

Command The target continues the Command phase and goes to the Message Out phase after the whole CDB is received.

Data In/Out The target recognizes the Attention Condition on block boundaries or at the end of the data phase. The target goes to the Message Out phase before going to any other phase. The target returns Check Condition status for the current command after servicing the Attention Condition unless the command is aborted during the Message Out phase.

Status The target goes to the Message Out phase after the Status byte is transferred.

Message In The target goes to the Message Out phase on a byte boundary between messages. The target goes to the Message Out phase before sending the next message.

Selection The target goes to the Message Out phase following the selection phase.

Should attention drop before the Identify message or between the Queue message and the message tag, then the target recognizes a protocol error and goes Bus Free.

Reselection The target goes to the Message Out phase after sending an Identify message in the Message In phase.

Supported messages

The following messages are supported.

Figure 73. SCSI message codes

Code	Message	Direction	Negate ATN Before Last ACK
06h	Abort Task Set	Out	Yes
0Dh	Abort Task	Out	Yes
24h	ACA	Out	No
16h	Clear ACA	Out	Yes
0Eh	Clear Task Set	Out	Yes
04h	Disconnect	In	N/A
21h	Head of Queue Tag (two bytes)	Out	No
80h-FFh	Identify	In	N/A
80h-FFh	Identify	Out	No
23h	Ignore Wide Residue	In	N/A
05h	Initiator Detected Error	Out	Yes
0Ah	Linked Command Complete	In	N/A
17h	LUN Reset	Out	Yes
09h	Message Parity Error	Out	Yes
07h	Message Reject	In	N/A
07h	Message Reject	Out	Yes
08h	No Operation	Out	Yes
22h	Ordered of Queue Tag (two bytes)	Out	No
20h	Simple Queue Tag (two bytes)	In	N/A
20h	Simple Queue Tag (two bytes)	Out	No
02h	Save Data Pointer	In	N/A
01h [01h]	Synchronous Data Transfer Request (Extended Message)	In	N/A
01h [01h]	Synchronous Data Transfer Request (Extended Message)	Out	Yes
0Ch	Target Reset	Out	Yes
00h	Task Complete	In	N/A
01h [03h]	Wide Data Transfer Request (Extended Message)	In	N/A
01h [03h]	Wide Data Transfer Request (Extended Message)	Out	Yes

Notes:

Key   Meaning
In   Target to Initiator
Out   Initiator to Target
Yes   Initiator shall negate ATN before last ACK of message
No   Initiator may or may not negate ATN before last ACK of message
[   ] Extended message

Message Out phase

The following messages are supported in the Message Out phase.

Some of the messages indicate that they must be the last message of the Message Out phase. This means that the initiator must negate the Attention signal after the target asserts Req and before the initiator asserts Ack for the last byte of the message. If the initiator fails to do this, the target goes Bus Free. The current task is aborted.

Abort Task (0Dh)

See Chapter 5, "Task management operations" on page 27 for a description of the task management function that is executed when this message is received.

This message must be the last message of the Message Out phase.

Abort Task Set (06h)

See Chapter 5, "Task management operations" on page 27 for a description of the task management function that is executed when this message is received.

This message must be the last message of the Message Out phase.

ACA (24h)

This message causes the command that is sent during this selection to be an "ACA command". See "Auto contingent allegiance condition" on page 20 for information about ACA commands.

This message must be the second message sent following a selection. It must follow an Identify message.

Clear ACA (16h)

See Chapter 5, "Task management operations" on page 27 for a description of the task management function that is executed when this message is received.

This message must be the last message of the Message Out phase.

Clear Task Set (0Eh)

See Chapter 5, "Task management operations" on page 27 for a description of the task management function that is executed when this message is received.

This message must be the last message of the Message Out phase.

Head of Queue

21h	Queue Tag

Queue Tag An unsigned 8 bit number.

See "Tagged queuing" on page 19 for a description of this message.

This message must be the second message sent following a selection. It must follow an Identify message.

Identify

Bit 7	Bit 6	Bits 5-0
1	Disconnect Privilege	LUN

Disconnect Privilege Disconnect Privilege=0 specifies that the target is not allowed to disconnect from the SCSI bus.

Disconnect Privilege=1 specifies that the target is allowed to disconnect from the SCSI bus.

LUN This field specifies which logical unit is being addressed with this connection.

If the initiator wants to send any Identify messages, it must send one immediately following the Selection phase. The initiator can send an Identify message at any time during a connection to change the Disconnect Privilege. The LUN values must be the same in all Identify messages sent during a connection. Otherwise, the target goes Bus Free. The current task is aborted.

Initiator Detected Error (05h)

The Initiator Detected Error message is sent from an initiator to inform the target that an error has been detected that does not preclude the target from retrying the previous Command, Data, or Status phase. The source of the error can either be related to previous activities on the SCSI bus or can be internal to the initiator and unrelated to any previous SCSI bus activity.

If the initiator intends to send this message, the initiator must assert the ATN signal prior to its release of Ack for the last byte transferred in the Information phase that is to be retried. This provides an interlock so the target can determine which Information phase to retry.

For most cases, the target aborts the current task and returns Check Condition status. The sense data is set to `Command Aborted - Initiator Detected Error Message Received`.

This message must be the last message of the Message Out phase.

LUN Reset (17h)

See Chapter 5, "Task management operations" on page 27 for a description of the task management function that is executed when this message is received.

This message must be the last message of the Message Out phase.

Message Parity Error (09h)

If this message follows a Message In phase, then the target retries the Message In phase starting with the message during which the initiator caused the Attention condition.

If this message follows any phase other than Message In phase, then the target goes Bus Free and aborts the current task.

This message must be the last message of the Message Out phase.

Message Reject (07h)

This message is sent from the initiator to indicate that the message it was receiving at the time it created the Attention condition was inappropriate or not supported. If this message follows a phase other than Message In phase, then the target goes Bus Free and aborts the current task. Otherwise, the target's action depends on the message that is being rejected.

Synchronous Data Transfer Request

Wide Data Transfer Request See "Wide/Synchronous negotiations" on page 87.

All Others Go Bus Free and abort the current task.

This message must be the last message of the Message Out phase.

No Operation (08h)

The target's action depends on which SCSI phase was interrupted by the Attention condition. If this message is received during the data phase the task will be terminated with Check Condition status. This message must be the last message of the Message Out phase.

Ordered

X'22'   Queue Tag

Queue Tag An unsigned 8 bit number.

See "Tagged queuing" on page 19 for a description of this message.

This message must be the second message sent following a selection. It must follow an Identify message.

Simple

X'20'   Queue Tag

Queue Tag An unsigned 8 bit number.

See "Tagged queuing" on page 19 for a description of this message.

This message must be the second message sent following a selection. It must follow an Identify message.

Synchronous Data Transfer Request

X'01'   X'03'   X'01'   Period4   Offset

Period4   This field is an unsigned 8 bit number.

The desired number of nanoseconds between Reqs or Acks divided by 4.

Offset   This field is an unsigned 8 bit number.

The desired number of Reqs that the target can send before waiting for an Ack signal. Offset=0 means that Asynchronous data transfer is desired.

See "Wide/Synchronous negotiations" on page 87 for information about Synchronous Data Transfer negotiation.

ESS responds to each initiator requested transfer period as shown in the following table:

Initiator Request (Mi)	Target Response (Mt)	Target Transfer Period
$0 \leq Mi \leq 12$	Mt = 12	50 nSec
$13 \leq Mi \leq 15$	Mt = Mi	62 nSec
$16 \leq Mi \leq 18$	Mt = Mi	75 nSec
$19 \leq Mi \leq 21$	Mt = Mi	87 nSec
$22 \leq Mi \leq 24$	Mt = Mi	99 nSec
$25 \leq Mi \leq 25$	Mt = 25	100 nSec
$26 \leq Mi \leq 31$	Mt = Mi	125 nsec
$32 \leq Mi \leq 37$	Mt = Mi	150 nsec
$38 \leq Mi \leq 43$	Mt = Mi	175 nsec

Initiator Request (Mi)	Target Response (Mt)	Target Transfer Period
44 ≤ Mi ≤ 49	Mt = Mi	199 nsec
50 ≤ Mi ≤ 50	Mt = 50	200 nsec
51 ≤ Mi ≤ 62	Mt = Mi	250 nsec
63 ≤ Mi ≤ 75	Mt = Mi	300 nsec
76 ≤ Mi ≤ 225	Mt = Mi	(Asynchronous mode)

This message must be the last message of the Message Out phase.

The target responds with an SDTR in the Message In phase.

Target Reset (0Ch)
See Chapter 5, "Task management operations" on page 27 for a description of the task management function that is executed when this message is received.

This message must be the last message of the Message Out phase.

Wide Data Transfer Request

X'01'	X'02'	X'03'	Transfer Width Exponent(m)

Transfer Width Exponent This field is an unsigned 8 bit number.

m=0   Data transfer at 8 bits wide is requested.

m=1   Data transfer at 16 bits wide is requested.

m=2   Data transfer at 32 bits wide is requested.

Initiator Request	Target Response	Target Data Transfer Width
Mi = 0	Mt = 0	8 Bit Data Transfers
Mi > 0	Mt = 1	16 Bit Data Transfers

See "Wide/Synchronous negotiations" on page 87 for information about Wide Data Transfer negotiation.

This message must be the last message of the Message Out phase.

The target responds with a WDTR in the Message In phase.

Message In phase
The target can send these messages in the Message In phase.

Disconnect (04h)
This message informs the initiator that the current connection is about to be ended. A subsequent Reselection is performed by the target for the current command. The target goes Bus Free following the successful transmission of this message.

Identify

Bit 7	Bit 6	Bits 5-0
1	Disconnect Privilege	LUN

LUN   This field specifies for which logical unit the target is reconnecting.

The target sends this messages immediately following every Reselection phase.

Ignore Wide Residue

X'23'   Residue

Residue   This field indicates the number of bytes that should be ignored. The bytes that should be ignored were transferred with the last Req of the Data In phase.

This field should be greater than 0 and less than the negotiated bus width (in bytes).

The target sends this message when it has completed a Data In phase for a number of bytes which is not divisible by the negotiated bus width.

Linked Command Complete (0Ah)

The target sends the Linked Command Complete message to the initiator to indicate that execution of a Linked command (with flag bit equal to zero) has completed and that valid status has been sent to the initiator. After successfully sending this message, the target changes to the Command phase to receive the next command.

Message Reject (07h)

The target sends this message if the last message received in the Message Out phase was inappropriate or not supported.

The target changes to the Message In phase immediately after receiving the offending message. The target does not wait for the Attention signal to be deasserted.

Save Data Pointer (02h)

The Save Data Pointer message is sent from the target to direct the initiator to save a copy of the present active data pointer. The value of the current (or active) data pointer should be moved into the corresponding saved data pointer for the currently attached logical unit.

The target always sends this message immediately prior to the Disconnect message.

Simple

X'20'   Queue Tag

Queue Tag An unsigned 8 bit number.

The target sends this message following the Identify message on Reselections for commands that have Queue tags associated with them.

Synchronous Data Transfer Request

X'01'   X'03'   X'01'   Period4   Offset

Period4   This field is an unsigned 8 bit number.

The desired number of nanoseconds between Reqs or Acks divided by 4.

Offset   This field is an unsigned 8 bit number.

The desired number of Reqs that the target can send before waiting for an Ack signal. Offset=0 means that Asynchronous data transfer is desired.

Initiator's Response (Mi)	Target Transfer Period
$0 \leq Mi \leq 11$	Send Message Reject (Asynchronous mode)
$12 \leq Mi \leq 12$	50 nSec
$13 \leq Mi \leq 15$	62 nSec
$16 \leq Mi \leq 18$	75 nSec
$19 \leq Mi \leq 21$	87 nSec
$22 \leq Mi \leq 24$	99 nsec
$25 \leq Mi \leq 25$	100 nSec
$26 \leq Mi \leq 31$	125 nSec
$32 \leq Mi \leq 37$	150 nSec
$38 \leq Mi \leq 43$	175 nSec
$44 \leq Mi \leq 49$	199 nSec
$50 \leq Mi \leq 50$	200 nSec
$51 \leq Mi \leq 62$	250 nSec
$63 \leq Mi \leq 75$	300 nSec
$76 \leq Mi \leq 225$	Send Message Reject (Asynchronous mode)

See "Wide/Synchronous negotiations" on page 87 for information about Synchronous Data Transfer negotiation.

Task Complete (00h)

This message indicates that the command has completed. The target goes Bus Free and releases all internal resources that were allocated to the task. This message is sent following the Status phase. It is sent whether the command completed successfully or failed.

Wide Data Transfer Request

X'01'	X'02'	X'03'	Transfer Width Exponent(m)

Transfer Width Exponent This field is an unsigned 8 bit number.

m=0   Data transfer at 8 bits wide is requested.

m=1   Data transfer at 16 bits wide is requested.

Initiator's Response	Target Data Transfer Width
Mi = 0	8 Bit Data Transfer
Mi = 1	16 Bit Data Transfer
Mi > 1	Send Message Reject (8 Bit Data Transfer)

See "Wide/Synchronous negotiations" on page 87 for information about Wide Data Transfer negotiation.

Wide/Synchronous negotiations

Negotiations for the width of the bus and the rate of data transfer are done on a target/initiator basis.

The target initiates negotiations on the first connection following a Reset, Target Reset message, Power On, or Self Initiated Reset. No data phases occur before the negotiations.

The initiator can initiate negotiations at any time. The target responds appropriately.

Wide Data Transfer negotiations must precede Synchronous Data Transfer negotiations. If an initiator initiates an SDTR negotiation only and the bus behavior allows the target to initiate negotiation, the target responds with a SDTR message and then reinitiates both negotiations. The target negotiates WDTR first and SDTR second. This renegotiation occurs before the next data phase.

If the initiator initiates a WDTR negotiation and does not initiate SDTR before the next data phase and the bus behavior allows the target to initiate negotiation, then the target reinitiates both negotiations.

The target responds to WDTR negotiations with a bus width that is not wider than the width offered by the initiator. The target responds to SDTR negotiations with a transfer rate that is not faster than the rate offered by the initiator.

If the initiator rejects an SDTR message from the target, then the target uses Asynchronous data transfers. If the initiator rejects a WDTR message from the target, then the target uses 8-bit wide data transfers, and negotiate synchronous.

For Fast-20 timings, a negotiated period of 48 ns is assumed to mean 50 ns.

Chapter 9. Fibre channel

ESS supports attachment using fibre channel using either point-to-point or Arbitrated Loop. An FC port is configured by the CE to be either point-to-point or Arbitrated Loop. Fabric attach (switch) is supported in the point-to-point configuration.

Fibre channel is logically a bi-directional serial data channel between two *nodes*. nodes are physically connected by a *link*; the point of connection between the link and the node is called a *port*.

Ports can be connected point-to-point by a single link or by a switching network (*fabric*). If the port is part of the fabric it is called an *F_port*, otherwise it is an *N_port*.

Fibre channel is defined in terms of a hierarchy of functions or *protocol layers*.

- FC-0: Physical Link
- FC-1: Transmission Protocol
- FC-2: Signaling and Framing Protocol
- FC-4: Mapping

All layers except FC-4 are defined in *Fibre Channel Physical and Signaling Interface (FC-PH)*

FC-4, the mapping layer, defines how other communication protocols (for example SCSI, IPI-3, HIPPI, FICON) can use fibre channel functions. The FC-4 used by ESS is FCP which is defined in *Fibre Channel Protocol for SCSI (SCSI-FCP)*

The following are the level of specifications used by ESS:

- FC-AL-1 rev 4.4

- FC-PH-3

- FCP-2

- PLDA 2.1

FC-AL

The Arbitrated Loop is an extension of FC-1 and FC-2 that allows more than two nodes to be connected without the expense of a fabric. A port that connects to an Arbitrated Loop is generally referred to as an *L_Port*. However, the terms *NL_Port* and *FL_Port* are also used if it is necessary to distinguish whether or not the L_Port is part of a fabric. Fibre Channel Arbitrated Loop (FC-AL) is defined in *Fibre Channel Arbitrated Loop (FC-AL).*

Another document, the 'Disk Profile', recommends which features from the other documents should and should not be implemented in order to ensure compatibility between devices from different manufacturers. The full title of the Disk Profile is *Fibre Channel Private LOOP SCSI Direct Attach (FC-PLDA)*

For ports configured as FC-AL ports, ESS intends to comply with the disk profile, which makes the following requirements.

89

1. ESS is a Private Loop device.

   - The loop may have an FL_Port, but:
     - No attempt is made to perform fabric Login
     - No attempt is made to communicate with the FL_Port
     - No response is given if the FL_Port attempts to communicate
   - Direct point-to-point attachment with an N_Port is not supported

2. The FC-4 is SCSI-FCP (see "FCP" on page 120).

3. ESS provides a Class 3 fibre channel service.

   - No indication of either successful or unsuccessful class 3 frame delivery is transmitted
   - Frame flow control is buffer-to-buffer only

Arbitrated Loop Physical Address (AL_PA)

An AL_PA is an 8-bit value which uniquely identifies an L_Port on the loop.

A hard AL_PA is one which is supplied through adapter VPD and provided during configuration by the CE. A soft AL_PA is one which is assigned during Loop Initialization.

FC - Point-to-Point

Point-to-point can attach to a single host or can attach using fabric to several hosts. These hosts can include S/390 hosts as well as Fixed Block hosts.

Point-to-point supports both class 2 and class 3 service.

Point-to-Point, port initialization

1. Start of Initialization. Whenever an N_Port receives an LR, LRR, OLS, or NOS Primitive Sequence, it begins this initialization process. The N_Port suspends all open Exchanges with all other N_Ports and NL_Ports. The N_Port shall proceed to step 2.

2. Implicit logout. If an OLS or NOS is received, or if LR, LRR, OLS, or NOS is received and the N_Port had not completed FLOGI prior to the start of initialization, the N_Port implicitly logged out from the fabric. The N_Port shall proceed to step 3.

3. Address authentication. After completion of Link Initialization, the N_Port performs FLOGI. After the FLOGI is completed successfully (the N_Port has received FLOGI ACC):

   a. If the N_Port has the same N_Port_ID, N_Port_Name, and Node_Name that it had before initialization, and the N_Port also determines that the F_Port has the same address, F_Port_Name, and Fabric_Name that the F_Port had before initialization, the N_Port resumes all suspended Exchanges with all other N_Ports. The N_Port shall proceed to step 5 on page 91.

   b. If the N_Port does not have the same N_Port_ID, N_Port_Name, or Node_Name that it had before initialization, or the N_Port determines that the F_Port does not have the same address, F_Port_Name, or Fabric_Name that the F_Port had before initialization, the N_Port discards all pending exchanges in a manner consistent with the ULP, implicitly

LOGO with all other N_Ports, and performs ULP-specific actions to clear pending tasks. The N_Port waits for R_A_TOV before originating any new exchanges. During this time, the N_Port discards all frames received except for the PLOGI and ADISC request sequences, and LOGO ACC reply sequences. The N_Port responds to all PLOGI request sequences with an LS_RJT reply sequence, with a reason code of "unable to perform command request at this time". The N_Port responds to all ADISC request sequences with a LOGO request sequence to the N_Port that initiated the PLOGI or ADISC request Sequence. The N_Port shall proceed to step 4.

4. Registration.

   a. The N_Port attempts to register with the fabric controller for state change notification (see "State Change Registration (SCR)" on page 116). If the attempted SCR fails the N_Port does not retry the attempt.

   b. The N_Port performs PLOGI explicitly with the Directory Server (address hex FFFFFC) and attempts an RFC-4 (see "Register FC-4 Type (RFC-4)" on page 120) request with the Name Service. If the attempted RFC-4 fails, the N_Port does not retry the attempt. The N_Port shall proceed to 5.

5. Initialization complete. This completes initialization for the N_Port. The port now allows PLOGIs from initiators.

FC-2 Protocol

Node and port names

ESS is a multiport storage facility. The port name comes from VPD on the adapter card. The node name is from product VPD.

Primitive signals and sequences

The fundamental unit of transfer on a fibre channel link is the 8b/10b encoded Transmission Character. Only 256 characters are required to represent a byte of data, so the set of valid transmission characters is sub-divided into Data Characters and Special Characters.

A *word* is a group of four consecutive transmission characters.

An *ordered set* is a word that starts with a special character (to give word and byte sync) and that has special significance to the communication protocol. fibre channel defines the following types of ordered sets:

- Frame delimiters
- Primitive signals
- Primitive sequences

A Frame Delimiter immediately precedes or follows the contents of a frame (see "Frames" on page 93). fibre channel defines the following:

SOF   Start of frame

EOF   End of frame

A primitive signal is an ordered set which has special meaning when received by itself. fibre channel defines the following:

| **R_RDY** Receiver Ready

| **IDLE** Idle

| Arbitrated Loop adds the following:

| **ARBx** Arbitrate

| **ARB(F0)** Arbitrate

| **OPNyx** Open Full-duplex

| **OPNyy** Open Half-duplex

| **OPNfr** Open Broadcast Replicate[4]

| **OPNyr** Open Selective Replicate[4]

| **CLS** Close

| **MRKtx** Mark[4]

| A *Primitive Sequence* is an ordered set that is transmitted repeatedly and
| continuously. Three or more of the ordered sets must be received consecutively in
| order to recognize the primitive sequence. fibre channel defines the following:

| **NOS** Not Operational[5]

| **OLS** Offline[5]

| **LR** Link Reset[5]

| **LRR** Link Reset Received[5]

| Arbitrated Loop adds the following:

| **LPByx** Loop Port Bypass[6]

| **LPEyx** Loop Port Enable

| **LPEfx** Loop Port Enable all

| **LIP(F7,F7)** Loop Initialization, no valid AL_PA

| **LIP(F8,F7)** Loop Initialization, loop failure; no valid AL_PA

| **LIP(F7,AL_PS)** Loop Initialization, valid AL_PA

| **LIP(F8,AL_PS)** Loop Initialization, loop failure; valid AL_PA

| **LIP(AL_PD,AL_PS)** Loop Initialization, reset L_Port.

| **LIP(FF,AL_PS)** Loop Initialization, reset L_Port; invalid or valid AL_PA if AL_PS
| does not refer to this port.

| [4] ESS retransmits but does not support

| [5] Used to convey information about a dedicated connection and therefore not relevant to Arbitrated Loop. If detected during an
| open connection, ESS immediately closes. Otherwise it simply retransmits.

| [6] Once bypassed, no FC operation or sequence will bring a port back online.

Frames

Information transfer is achieved using frames that are constructed from words and ordered sets. All frames have the same general format, as shown in Figure 74.

```
Bytes    (4)       (24)      (0 to 2048)    (4)    (4)

        ┌─────┬────────┬──────────────┬──────┬─────┐
        │ SOF │ Header │   Payload    │ CRC  │ EOF │
        └─────┴────────┴──────────────┴──────┴─────┘
        ◄──────────────────────────────►
                                         Payload      : :
                                                      : :
                                         Fill bytes
                                         (if any)
```

Figure 74. General frame format

The SOF ordered set indicates the start of frame and provides word synchronization.

The header is the first field after the SOF delimiter. It is used by the link control facility to control link operations, control device protocol transfer, and to detect missing or out of order frames. The header is shown in Figure 75.

FC-PH describes the content of each field except 'Parameter'. SCSI-FCP defines this field as RLTV_OFF (Relative Offset).

```
        Bits

      ┌──────────┬──────────┬──────────┬──────────┐
      │3322 2222 │2222 1111 │1111 11   │          │
 Word │1098 7654 │3210 9876 │5432 1098 │7654 3210 │
      ├──────────┼──────────┴──────────┴──────────┤
   0  │  R_CTL   │             D_ID                │
      ├──────────┼─────────────────────────────────┤
   1  │ Reserved │             S_ID                │
      ├──────────┼─────────────────────────────────┤
   2  │  TYPE    │             F_CTL               │
      ├──────────┼──────────┬──────────────────────┤
   3  │  SEQ_ID  │  DF_CTL  │       SEQ_CNT        │
      ├──────────┴──────────┼──────────────────────┤
   4  │      OX_ID          │       RX_ID          │
      ├─────────────────────┴──────────────────────┤
   5  │               Parameter                    │
      └─────────────────────────────────────────────┘
```

Figure 75. Frame header

The payload follows the header and has a length 0 - 2048 bytes that is divisible by 4.

The cyclic redundancy check (CRC) is a four-byte field following the payload. It is used to verify the integrity of the header and payload.

The EOF ordered set marks the end of frame.

Sequences

A sequence is a set of one or more related frames that flow in one direction only. The sequence is identified by the Sequence Identifier (SEQ_ID) field in the frame header.

Exchanges

An exchange is a set of one or more related non-concurrent sequences that can flow in the same or opposite directions. The exchange is identified by an Originator Exchange Identifier (OX_ID) and a Responder Exchange Identifier (RX_ID) in the frame header.

Basic Link services

The Basic Link services, with the exception of BA_ACC and BA_RJT, have no payload. The header Type field is set to 00h (Basic Link Service) and R_CTL is set to 1000xxxxb (Basic Link_Data, Command=xxxx).

Figure 76. Basic Link Service command codes

Code	Description	Abbreviation
0h	No Operation	NOP
1h	Abort Sequence	ABTS
2h	Remove Connection	RMC
4h	Basic Accept	BA_ACC
5h	Basic Reject	BA_RJT
Others	Reserved	

NOP and RMC are prohibited by the FC-PLDA. If received, the response is a BA_RJT with reason code 01 and reason explanation 00.

Abort Sequence (ABTS)

Although ABTS is a fibre channel Basic Link service, it is used by SCSI-FCP to implement the Abort Task, Task Management function. It can only be used by a SCSI initiator to abort an entire exchange using the Recovery Abort protocol.

The response to ABTS is either BA_ACC or BA_RJT.

Basic Accept (BA_ACC)

BA_ACC indicates that a Basic Link service request has been completed.

Figure 77. BA_ACC payload

Byte	Item	Size (Bytes)
0	SEQ_ID Validity	1
1	SEQ_ID	1
2-3	Reserved	2
4-5	OX_ID	2
6-7	RX_ID	2
8-9	Low SEQ_CNT	2
10-11	High SEQ_CNT	2

Basic Reject (BA_RJT)

BA_RJT indicates that a Basic Link service request has been rejected. The payload contains a four byte reason code that indicates why the request was rejected.

Figure 78. BA_RJT payload

Byte	Item	Size (Bytes)
0	Reserved	1
1	Reason Code	1
2	Reason Explanation	1
3	Vendor Unique	1

Figure 79. BA_RJT reason codes

Code	Description
01h	Invalid command code
03h	Logical error
05h	Logical busy
07h	Protocol error
09h	Unable to perform command requested
FFh	Vendor unique error
Others	Reserved

Figure 80. BA_RJT reason code explanations

Code	Description
00h	No additional explanation
03h	Invalid OX_ID-RX_ID combination
05h	Sequence aborted, no sequence information provided
Others	Reserved

The meaning of the BA_RJT reason codes are:

Invalid Command Code
> The command code in the sequence being rejected is invalid.

Logical Error The request identified by the command code is invalid or logically inconsistent for the conditions present.

Logical busy The Basic Link service is logically busy and unable to process the request at this time.

Protocol Error This indicates that an error has been detected that violates the rules of FC-2 protocol, which are not specified by other error codes.

Unable to perform command request
> ESS is unable to perform the request at this time.

Vendor Unique Error
> Not used.

Extended Link services

For Extended Link service frames, the header Type field is set to 01h (Extended Link Service). R_CTL is either 22h (Extended Link Data, Unsolicited Control) for a request, or to 23h (Extended Link Data, Solicited Control) for a reply.

The first byte of the payload is the LS_Command and encodes the Request or Reply, as shown in Figure 81 and Figure 82.

Figure 81. Extended Link Service replies

Code	Reply	Abbreviation
02h	Accept	ACC
01h	Link Service Reject	LS_RJT

Figure 82 is a comprehensive list of all of the Extended Link service commands supported by ESS.

Figure 82. Extended Link service requests

Code	Request	Abbreviation
52h	Address Discovery	ADISC
10h	Echo	ECHO
04h	Fabric Login[7]	FLOGI
05h	Logout	LOGO
50h	Port Discovery	PDISC
03h	Port Login	PLOGI
20h	Process Login	PRLI
21h	Process Logout	PRLO
0Fh	Read Link Status	RLS
0Eh	Read Timeout Value	RTV
61h	Registered State Change Notification[7]	RSCN
53h	Report Node Capabilities	RNC
12h	Reinstate Recovery Qualifier	RRQ
62h	State Change Registration[7]	SCR
24h	Third Party Process Logout	TPRLO

Note: Only lists extended link services (ELS) supported by ESS. Refer to FC-PH for a complete list of ELSs. Any Extended Link service requested that is not in this table will be responded to by an LS_RJT with a reason code of 0B, and a reason code explanation of 00h.

---

[7] Only valid in point-to-point configurations.

Accept (ACC)

ACC is used in response to an Extended Link service request. It indicates that the request has been completed.

The ACC payload depends upon the Extended Link service request and is therefore described separately for each of the following:

- "ECHO (ECHO)" on page 98
- "Port Login (PLOGI)/Fabric Login (FLOGI)" on page 98
- "Logout (LOGO)" on page 104
- "Port Discovery (PDISC)" on page 104
- "Address Discovery (ADISC)" on page 105
- "Process Login (PRLI)" on page 106
- "Process Logout (PRLO)" on page 109
- "Read Timeout Value" on page 112
- "Read Link Error Status Block (RLS)" on page 111
- "Registered State Change Notification (RSCN)" on page 113
- "Report Node Capabilities (RNC)" on page 114
- "State Change Registration (SCR)" on page 116
- "Re-instate Recovery Qualifiers (RRQ)" on page 115
- "Third Party Process Logout (TPRLO)" on page 116

Link Service Reject (LS_RJT)

LS_RJT indicates that the Extended Link service request has been rejected. The payload (shown in Figure 83) contains a reason code and a reason explanation.

Figure 83. LS_RJT payload

Byte	Item	Size (Bytes)
0-3	0100 0000h	4
4	Reserved	1
5	Reason Code (see Figure 84)	1
6	Reason Explanation (see Figure 85 on page 98)	1
7	Vendor Unique	1

Figure 84. LS_RJT reason codes

Code	Description
Others	Reserved
01h	Invalid Command Code
03h	Logical Error
05h	Logical Busy
07h	Protocol error
09h	Unable to perform command requested
0Bh	Command not supported
FFh	Vendor unique error

Figure 85. LS_RJT reason code explanations

Code	Description
00h	No additional explanation
01h	Service Parm error - Options
03h	Service Parm error - Initiator Ctl
05h	Service Parm error - Recipient Ctl
07h	Service Parm error - Rec Data Field Size
09h	Service Parm error - Concurrent Sequence
0Bh	Service Parm error - Concurrent Credit
0Dh	Invalid Port name
0Eh	Invalid Node/Fabric Names
0Fh	Invalid Common Service Parameters
19h	Command (request) already in progress
1Fh	Invalid N_Port identifier
29h	Insufficient resources to support Login
2Ah	Unable to supply requested data
2Ch	Request not supported

Note: Refer to FC-PH for a full list. Only the explanations relevant to supported Extended Link services are shown.

ECHO (ECHO)

The Echo extended link service request sequence is a single frame requesting the recipient to transmit the payload contents, following the LS_Command, back to the initiator of the Echo command in the same order as received using the ACC reply sequence consisting of a single frame. The Echo Extended Link service request provides a means to transmit a data frame and have the payload content returned for a simple loop-back diagnostic function. The Echo command is transmitted as a one frame sequence. The ACC reply sequence is also a one frame sequence.

Figure 86. ECHO payload

Byte	Item	Size (Bytes)
0-3	1000 0000h	4
4-nn	ECHO data	Max Frame

Figure 87. ECHO accept payload

Byte	Item	Size (Bytes)
0-3	0200 0000h	4
4-nn	ECHO data	= Echo

Port Login (PLOGI)/Fabric Login (FLOGI)

PLOGI is used by the initiator to register service parameters with the target. The target responds with an ACC that has a payload similar to the PLOGI but that contains the target's parameters. This exchange of parameters establishes the operating environment between the initiator and the target.

FLOGI is used by ESS to register service parameters with the fabric controller. The fabric controller responds with an ACC that has a payload similar to the FLOGI but that contains the target's parameters. This exchange of parameters establishes the operating environment between the N_Port and the F_Port. ESS initiates and accepts FLOGI when in point-to-point mode. ESS rejects FLOGI with an LS_RJT, with a reason code of OBh and a reason code explanation of 00h, when in Arbitrated Loop mode.

Bytes 1, 2, and 3, of the payload must be set to zero. Otherwise, ESS responds with an LS_RJT, with a reason code of 01h and a reason code explanation of 00h.

Figure 88. PLOGI/FLOGI payload

Byte	Item	Size (Bytes)
0-3	0300 0000h (PLOGI) 0400 0000h (FLOGI)	4
4-19	Common Service Parameters	16
20-27	Port Name	8
28-35	Node Name	8
36-51	Class 1 Service Parameters	16
52-67	Class 2 Service Parameters	16
68-83	Class 3 Service Parameters	16
84-99	Reserved	16
100-115	Vendor Version Level	16

Figure 89. PLOGI ACC payload

Byte	Item	Size (Bytes)
0-3	0200 0000h	4
4-19	Common Service Parameters	16
20-27	Port Name	8
28-35	Node Name	8
36-51	Class 1 Service Parameters	16
52-67	Class 2 Service Parameters	16
68-83	Class 3 Service Parameters	16
84-99	Reserved	16
100-115	Vendor Version Level	16

Common Service Parameters: The common service parameters apply to all classes of service and are exchanged during login.

Figure 90. Common Service Parameters

Byte	Item	Size (Bytes)	ESS Response
0-1	FC-PH Version	2	20 08
2-3	BB_Credit (PT-PT)	2	00 40
	BB_Credit (FC-AL)		00 00
4-5	Common Features (PT-PT)	2	80 00
	Common Features (FC-AL)		88 00
6-7	Buffer to buffer receive Data Field size	2	08 00
8-9	Total concurrent sequences	2	00 FF
10-11	Relative offset by info category	2	FF FF
12-15	E_D_TOV	4	0000 2710. This value is replaced by the E_D_TOV specified during the F_LOGI when fabric attached.

The upper byte of the FC-PH Version field indicates the highest version of FC-PH that is supported and the lower byte indicates the lowest. The code/version relationship is given by Figure 91. ESS checks to ensure that the payload for this field is a valid entry (shown in Figure 91), and responds with an LS_RJT if it is not.

Figure 91. FC-PH Version

Hex Value	Version
00	None
06	FC-PH 4.0
07	FC-PH 4.1
08	FC-PH 4.2
09	FC-PH 4.3
10h	FC-PH-2
20h	FC-PH-3
Others	Reserved

The BB_Credit field indicates the number of frame buffers available in the originator to hold frames from the receiver.

The Common Features field contains the following flags.

Bit 15 - Continuously Increasing Offset
When set (1), this flag indicates that the port supports continuously increasing relative offset within a sequence on a frame by frame SEQ_CNT basis. The relative offset is only present if bit 3 of the F_CTL field in the frame header is set to 1b. ESS returns a 1b in this field.

Bit 14 - Random Relative Offset
When set (1), this flag indicates that the port supports random relative offset within a sequence. The relative offset is only present if bit 3 of

the F_CTL field in the frame header is set to 1b. ESS returns a 0b in this field.

Bit 13 - Valid Vendor Version Level
When set (1), this flag indicates that the vendor version level field in the frame payload contains valid information. ESS returns a 0b in this field.

Bit 12 - N_Port/F_Port
When set (1), this flag indicates that the port is an F_Port. When reset (0), it is an N_Port. ESS returns a 0b in this field.

Bit 11 - Alternate BB-Credit Model
When set (1), this flag indicates that the port supports the Alternate BB-Credit model.

For FC-AL (per PLDA), this bit must be set (1) in the payload from the initiator. If not, ESS responds with an LS_RJT, and with a reason code of 03h and a reason code explanation of 0Bh. ESS returns a 1b in this field.

For point-to-point ESS returns a 0b in this field.

Bits 10-0
Reserved

The Buffer to Buffer Receive Data Field size indicates the largest frame payload (in bytes) that can be received by the port. Sizes that are less than 128, greater than 2112, or are not divisible by 4 are invalid.

The Total Concurrent Sequences. field indicates the number of concurrent sequences that the port can support for all 3 classes. It must be less than or equal to the sum of the concurrent sequences supported on a class by class basis.

The Relative Offset by Information Category field is all flags. The bit position of each set (1) bit determines the information category. For example, if relative offset is supported for category 0001b (solicited data), bit 1 is set.

The E_D_TOV field specifies the E_D_TOV (error detect timeout value) in units of 1 ms.

Class Service Parameters

┌──────────────── Point-to-Point only ────────────────┐

Because ESS supports classes 2 and 3 service, it only checks and responds to the class 2 and class 3 service parameters.

└──────────────── End of Point-to-Point only ────────────────┘

┌──────────────── FC-AL Only ────────────────┐

Because ESS supports class 3 service, it only checks and responds to the class 3 service parameters.

└──────────────── End of FC-AL Only ────────────────┘

Figure 92. Class Service Parameters

Byte	Item	Size (Bytes)	ESS Response
0-1	Service options	2	80 00
2-3	Initiator Control	2	00 00
4-5	Recipient Control - Class 2	2	80 00
	Recipient Control - Class 3		00 00
6-7	Receive Data Field size	2	08 00
8-9	Concurrent sequences	2	00 FF
10-11	End to end credit - class 2	2	00 0C
	End to end credit - class 3		00 00
12-13	Open Sequences per Exchange	2	00 01
14-15	Reserved	2	00 00

The Service Options field contains the following flags.

Bit 15 - Class Validity
> When set (1), this flag indicates that the class of service is supported. ESS supports class 2 and 3, therefore this bit is only set for the class 2 and 3 service parameters. If this bit is not set for class 3 service parameters, ESS responds with an LS_RJT, and with a reason code of 03h and a reason code explanation of 01h.

Bit 14 - Intermix Mode
> Only applies to class 1 service. Reserved for classes 2 and 3.

Bit 13 - Transparent Mode
> Does not apply to PLOGI.

Bit 12 - Lock Down Mode
> Does not apply to PLOGI.

Bit 11 - Sequential Delivery
> Does not apply to PLOGI. ESS returns 0b in this field in response to a PLOGI or FLOGI.
>
> When initiating a FLOGI, ESS sets this field to 1b. ESS must have sequential delivery in a switched environment.

Bits 10-0
> Reserved

The Initiator Control flags specify which protocols, policies or functions the supplier of the service parameters requests of the recipient or is capable of.

Bits 15-14 - X_ID Reassignment

> **0 0**   X_ID Reassignment not supported
>
> **0 1**   X_ID Reassignment supported
>
> **1 0**   Reserved
>
> **1 1**   X_ID Reassignment required

Bits 13-12 - Initial Process Associator

| | **0 0** | Initial Process Associator not supported |

| | **0 1** | Initial Process Associator supported |

| | **1 0** | Reserved |

| | **1 1** | Initial Process Associator required |

ESS accepts values of 00 and 01 in this field. If other values are sent, ESS responds with an LS_RJT, and with a reason code of 03h and a reason code explanation of 03h.

Bit 11 - ACK_0 Capability

Only applies to class 1 or class 2.

Bit 10 - ACK_N Capability

Only applies to class 1 or class 2.

Bits 9-0

Reserved

The Recipient Control flags specify which functions are supported by the supplier of the service parameters when receiving frames.

Bit 15 - ACK_0 Capability

Only applies to class 1 or class 2.

Bit 14 - ACK_N Capability

Only applies to class 1 or class 2.

Bit 13 - X_ID Interlock

Only applies to class 1 or class 2.

Bits 12-11 - Error Policy Supported

| | **0 0** | Only discard supported |

| | **0 1** | Reserved |

| | **1 0** | Both discard and process supported |

| | **1 1** | Reserved |

ESS checks to ensure that the reserved values are not sent (that is, 01 or 11). If they are, ESS responds with an LS_RJT, and with a reason code of 03h and a reason code explanation of 05h.

Bits 10-9 - Categories per Sequence

| | **0 0** | 1 Category/Sequence |

| | **0 1** | 2 Categories/Sequence |

| | **1 0** | Reserved |

| | **1 1** | More than 2 Categories/Sequence |

Bits 8-0

Reserved

The Receive Data Field size is the largest payload (in bytes) that the port is capable of receiving. It must be less than or equal to the buffer to buffer receive data field specified in the common service parameters. Sizes less than 128, greater than 2112, or not divisible by 4 are invalid, in which case ESS responds with an LS_RJT, and with a reason code of 03h and a reason code explanation of 07h.

The Concurrent Sequences field specifies the number of separate sequences that ESS is capable of tracking. The value sent must be greater than 0, or ESS responds with an LS_RJT, and with a reason code of 03h and a reason code explanation of 09h.

The End-to-end Credit field applies only to class 1 and class 2.

The Open Sequences per Exchange field indicates the maximum number of sequences per exchange that can be open at the recipient at one time. The value sent must be greater than 0, or ESS responds with an LS_RJT, and with a reason code of 03h and a reason code explanation of 00h.

Logout (LOGO)

LOGO is used by an initiator to request invalidation of the service parameters that were exchanged during PLOGI.

Bytes 1, 2, and 3 of the payload must be set to zero. Otherwise, ESS responds with an LS_RJT, with a reason code of 01h, and a reason code explanation of 00h.

Figure 93. LOGO payload

Byte	Item	Size (Bytes)
0-3	0500 0000h	4
4	Reserved	1
5-7	Port Identifier	3
8-15	Port Name	8

Figure 94. LOGO ACC payload

Byte	Item	Size (Bytes)
0-3	0200 0000h	4

Port Discovery (PDISC)

PDISC is used by an initiator to exchange service parameters without affecting the operating environment between the initiator and ESS. The PDISC and corresponding ACC payloads are exactly as defined for PLOGI (see "Port Login (PLOGI)/Fabric Login (FLOGI)" on page 98), except that byte 0 of the payload is 50h. It can be used as a means of authentication following a Loop Initialization process. If the port ID's node name and port name have not changed since the previous login, the initiator and target can continue where they left off prior to the LIP and all open exchanges continue. If the Port ID's node name and port name do not match the login values, the initiator is implicitly logged out by ESS.

If some means of authentication following a LIP does not occur within RR_TOV, the initiator is implicitly logged out by ESS.

Bytes 1, 2, and 3 of the payload must be set to zero. Otherwise, ESS responds with an LS_RJT, with a reason code of 01h and a reason code explanation of 00h.

Address Discovery (ADISC)

The ADISC ELS allows communicating N-Ports to exchange addresses and port/node name identifiers. It can be used as a quick means of authentication following a Loop Initialization process. If the Port ID's node name and port name have not changed since the previous login, the initiator and target can continue where they left off prior to the LIP, and all open exchanges continue. If the Port ID's node name and port name do not match the login values, the initiator is implicitly logged out by ESS. ADISC can also be used as a means to determine if a port was able to acquire its hard AL_PA during LIP.

Bytes 1, 2, and 3 of the payload must be set to zero. Otherwise, ESS responds with an LS_RJT, with a reason code of 01h and a reason code explanation of 00h.

If some means of authentication following a LIP does not occur within RR_TOV, the initiator is implicitly logged out by ESS.

The following tables show the ADISC payload and the ADISC ACC payload.

Figure 95. ADISC payload

Byte	Item	Size (Bytes)
0-3	Command Code = 52000000h	4
4	Reserved	1
5-7	Hard Address of Originator	3
8-15	Port Name of Originator	8
16-23	Node Name of Originator	8
24	Reserved	1
25-27	N_Port ID of Originator	3

Figure 96. ADISC ACC payload

Byte	Item	Size (Bytes)
0-3	ACC Code = 02000000h	4
4	Reserved	1
5-7	Hard Address of Responder	3
8-15	Port Name of Responder	8
16-23	Node Name of Responder	8
24	Reserved	1
25-27	N_Port ID of Responder	3

Hard Address:

- For Arbitrated Loop, this 3-byte identifier consists of:

  - The MSB is an 8 bit domain address. For private devices this should be zeroes ESS returns zeroes.
  - The middle byte is an 8 bit area address. For private devices this should be zeroes ESS returns zeroes.
  - The LSB is the 8 bit AL-PA that the port attempts to acquire during the LIHA sequence of LIP. For ESS, this number is from VPD for the port, and is entered by the CE.

When the Hard Address field is equal to the N_Port ID field, the port was able to obtain its hard address during LIP.

- For point-to-point, this is the last address known to this port.

The value sent from the originator is not checked by ESS.

Port Name: IEEE unique address. See "Node and port names" on page 91.

Node Name: IEEE unique address. See "Node and port names" on page 91.

N-Port ID: This is the 24 bit NL_Port Identifier used in the S_ID of the ADISC Accept header.

Process Login (PRLI)

The PRLI request informs the recipient of the capabilities and requirements of the originator. The recipient responds with an ACC to indicate agreement or LS_RJT otherwise. ESS only supports a single service parameter page with PRLI. If an attempt to write more than one service parameter page is detected ESS responds with reason code 07.

Figure 97. PRLI payload

Byte	Item	Size (Bytes)
0	20h	1
1	Page Length=10h	1
2-3	Payload Length=14h	2
4-19	Login service parameter page	16

Figure 98. PRLI ACC payload

Byte	Item	Size (Bytes)
0	02h	1
1	Page Length=10h	1
2-3	Payload Length=14h	2
4-19	Login response service parameter page	16

Process Login Service Parameter Page

Figure 99. Login Service Parameter page

Byte	Item	Size (Bytes)
0	Type Code = 08h	1
1	Type Code Extension = 00h	1
2-3	Flags	2
4-7	Originator Process Associator	4
8-11	Responder Process Associator	4
12-15	Service Parameters	4

The Type code field for SCSI-FCP is 08h and the Type Code Extension field is 00h.

The Flags field is as follows:

Bit 15 - Originator Process Associator Valid
This bit must be set to zero marking the originator process associator field of this service parameter page as invalid. If this bit is set to one ESS returns 04h in the accept response code field.

Bit 14 - Responder Process Associator Valid
This bit must be set to zero marking the responder process associator field of this service parameter page as invalid. If this bit is set to one ESS returns 04h in the accept response code field.

Bit 13 - Establish Image Pair
This bit must be set to 1 indicating that the originator wants to establish an image pair. If this bit is set to zero ESS returns 03h in the accept response code field.

Bits 12-0
Reserved

The Originator Process Associator identifies a group of related processes (an image) within the originator.

The Responder Process Associator identifies a group of related processes (an image) within the responder.

The Service Parameters field contains the following flags:

Bits 31-7
Reserved

Bit 6 - Data Overlay Allowed
When set (1), indicates that the initiator function is capable of supporting data overlay.

Bit 5 - Initiator Function
When set (1), indicates that the process defined by this page is operating as a SCSI Initiator.

This bit must be set (1) in order for ESS to accept the login request. If not, ESS responds with an LS_RJT, with a reason code of 03h and a reason code explanation of 00h.

Bit 4 - Target Function
When set (1), indicates that the process defined by this page is operating as a SCSI target.

Bit 3 - Command/Data Mixed Allowed
When set (1), indicates that FCP_CMND and FCP_DATA can be combined in one IU.

Bit 2 - Data/Response Mixed Allowed
When set (1), indicates that FCP_DATA and FCP_RSP can be combined in one IU.

Bit 1 - Read XFER_RDY Disabled
When set (1), indicates that the FCP_XFER_RDY IU cannot be used for SCSI read operations.

Bit 0 - Write XFER_RDY Disabled

> When set (1), indicates that the FCP_XFER_RDY IU cannot be used for the first write data of SCSI write operations.

Process Login Response Service Parameter Page

Figure 100. Login Response Service Parameter page

Byte	Item	Size (Bytes)	ESS Response
0	Type Code	1	08
1	Type Code Extension	1	00
2-3	Flags	2	21 XX
4-7	Originator Process Associator	4	00 00 00 00
8-11	Responder Process Associator	4	00 00 00 00
12-15	Service Parameters	4	00 00 00 13

The Type code field for SCSI-FCP is 08h and the *TYPE code Extension* field is 00h.

The Flags field is as follows:

Bit 15 - Originator Process Associator Valid

> Because ESS does not support process associators, this bit must be reset (0) in the payload. If not, ESS responds with a PRLI ACC response code of 07h.

Bit 14 - Responder Process Associator Valid

> Because ESS does not support process associators, this bit must be reset (0) in the payload. If not, ESS responds with a PRLI ACC response code of 04h.

Bit 13 - Image Pair Established

Bit 12 - Reserved

Bits 11-8 - Accept Response Code
> See Figure 101 on page 109.

Bits 7-0 - Reserved

The Originator Process Associator and Responder Process Associator fields are not used and are set to 00000000h by ESS.

The Service Parameters field contains the following flags:

Bits 31-6 - Reserved
> Set to zero

Bit 5 - Initiator Function
> Set to zero

Bit 4 - Target Function
> Set to one

Bit 3 - Command/Data Mixed Allowed
> Set to zero

Bit 2 - Data/Response Mixed Allowed
> Set to zero

Bit 1 - Read XFER_RDY Disabled
> Set to one

Bit 0 - Write XFER_RDY Disabled
> ESS defaults this field to 1b, if configured as point-to-point and if the initiator allows it.

Figure 101. PRLI/PRLO ACC response codes

Code	Description
00h	Reserved
01h	Request executed
02h	The target image has no resources available for establishing image pairs between the specified source and destination N_Ports. The PRLI request may be retried.
03h	Initialization is not complete for the target image. The PRLI request may be retried.
04h	The target image corresponding to the responder PA specified in the PRLI request and PRLI accept does not exist. The PRLI request is not retried.
05h	The target image has a predefined configuration that precludes establishing this image pair. The PRLI request is not retried.
06h	Request executed conditionally. Some service parameters were not able to be set to their requested state. See the service parameters response field for further details.
07h	The destination N_port is unable to process multiple page PRLI requests. The PRLI request can be retried as a single page request.
08h-FFh	Reserved

Process Logout (PRLO)

The PRLO request indicates to the responder that those process image pairs specified in the service parameter pages are being discontinued by the originator. All tasks, reservations, and status for the specified image pairs are set to the state they would have after a SCSI hard reset or power-on reset.

Figure 102. PRLO payload

Byte	Item	Size (Bytes)
0	21h	1
1	Page Length=10h	1
2-3	Payload Length = 14h	2
4-19	Logout service parameter page	16

Figure 103. PRLO ACC payload

Byte	Item	Size (Bytes)
0	02h	1
1	Page Length=10h	1
2-3	Payload Length=14h	2
4-19	Logout service parameter response page	16

Process Logout Service Parameter Page

Figure 104. Logout Service Parameter page

Byte	Item	Size (Bytes)
0	Type Code = 08h	1
1	Type Code Extension = 00h	1
2-3	Flags	2
4-7	Originator Process Associator	4
8-11	Responder Process Associator	4
12-15	Reserved	4

The Type code field for SCSI-FCP is 08h and the TYPE Code Extension field is 00h.

The Flags field is as follows:

Bit 15 - Originator Process Associator Valid
When set (1), the originator process associator field of this service parameter page is valid.

Bit 14 - Responder Process Associator Valid
When set (1), the responder process associator field of this service parameter page is valid.

Bits 13-0 - Reserved

The Originator Process Associator identifies a group of related processes (an image) within the originator.

The Responder Process Associator identifies a group of related processes (an image) within the responder.

Process Logout Response Service Parameter Page

Figure 105. Logout Response Service Parameter page

Byte	Item	Size (Bytes)	ESS Response
0	Type Code	1	08
1	Type Code Extension	1	00
2-3	Flags	2	0X 00
4-7	Originator Process Associator	4	00 00 00 00
8-11	Responder Process Associator	4	00 00 00 00
12-15	Reserved	4	00 00 00 00

The Type code field for SCSI-FCP is 08h and the Type Code Extension field is 00h.

The Flags field is as follows:

Bit 15 - Originator Process Associator Valid
> ESS returns 0b in this field.

Bit 14 - Responder Process Associator Valid
> ESS returns 0b in this field.

Bits 13-12 - Reserved

Bits 11-8 - Accept Response Code
> See Figure 101 on page 109.

Bits 7-0 - Reserved

The Originator Process Associator and Responder Process Associator fields are not used.

Read Link Error Status Block (RLS)

RLS requests the recipient to return the link error status block associated with the port identifier specified in the payload. ESS implements a link error status block for each port. When a counter overflows, it wraps back to zero. The only way to reset the link error status block is to power off ESS.

Bytes 1, 2, and 3 of the payload must be set to zero. Otherwise, ESS responds with an LS_RJT, with a reason code of 01h and a reason code explanation of 00h. valid port identifiers are:

0   Return the link error status block for the same port the request was received on.

Any other value for the port identify is rejected with an LS_RJT with reason code of "Invalid N_Port Identifier".

Figure 106. RLS payload

Byte	Item	Size (Bytes)
0-3	0F00 0000h	4
4	Reserved	1
5-7	Port Identifier	3

Figure 107. RLS ACC payload

Byte	Item	Size (Bytes)
0-3	0200 0000h	4
4-27	Link Error Status Block	24

Figure 108. Link Error Status Block

Byte	Item	Size (Bytes)
0-3	Link Failure Count	4
4-7	Loss of sync count	4
8-11	Loss of signal count	4
12-15	Primitive Sequence Protocol error	4
16-19	Invalid Transmission Word	4
20-23	Invalid CRC Count	4

ESS link error status block is defined as follows:

Link Failure Count: This is a count of the number of times that the port's receiver loses synchronization for a continuous period of time determined by RT_TOV. When a link failure condition is detected on a port, it generates NOS if in point-to-point, or transmits LIPF8 if Arbitrated Loop.

Loss of sync count: This is a count of the number of times that the port's receiver loses synchronization.

Loss of signal count:

Primitive Sequence Protocol error:

Invalid Transmission Word:

Invalid CRC Count:

Read Timeout Value

RTV requests that the recipient return the Resource_Allocation_Timeout Value (R_A_TOV) and the Error_Detect_Timeout Value (E_D_TOV) in the Accept reply sequence.

Bytes 1, 2, and 3 of the payload must be set to zero. Otherwise, ESS responds with an LS_RJT, with a reason code of 01h and a reason code explanation of 00h.

Figure 109. RTV payload

Byte	Item	Size (Bytes)
0-3	0E00 0000h	4

Figure 110. RTV ACC payload

Byte	Item	Size (Bytes)
0-3	0200 0000h	4
4-7	R_A_TOV	4
8-11	E_D_TOV	4
12-15	Timeout Qualifier	4

R_A_TOV Resource_Allocation_Timeout Value, see "FC Timers" on page 119

E_D_TOV Error_Detect_Timeout Value, see "FC Timers" on page 119

Timeout Qualifier Defined as follows:

Bit   Definition

31-27 Reserved

26   E_D_TOV time is in ns, otherwise if bit is zero E_D_TOV is in ms.

25-0 Reserved

Registered State Change Notification (RSCN)

A registered state change notification (RSCN) extended link service request is sent to registered N_Ports and NL_Ports when an event occurs which may have affected the state of an N_Port or NL_Port, or the ULP *state* within the N_Port or NL_Port. The term state is used here to refer to any condition of an N_Port or NL_Port that is considered important enough to notify other N_Ports and NL_Ports of a change in that state. The definition of important is specific to an N_Port or RSCN is intended to provide a timely indication of changes in nodes to avoid the considerable traffic that polling can generate.

RSCN can be used to detect a failed node and release resources tied up by the failed node. RSCN can also be used to notify interested nodes of new devices coming online, and of changes within an online node that affect the operation of the system (for example, more storage has become available). The sender of the RSCN request can coalesce several events into a single report.

The normal use of an RSCN request is for the fabric to notify registered N_Ports and NL_Ports of changes detected by the fabric. The RSCN request is issued by the fabric controller (hex FFFFFD) to the registered N_Ports and NL_Ports. An RSCN request can also be issued by an N_Port or NL_Port to the fabric controller to indicate changes of state within the N_Port or NL_Port that are not otherwise detectable by the fabric. In either case, the payload of an RSCN request includes a list containing the addresses of the affected N_Ports and NL_Ports.

Figure 111. RSCN payload

Byte	Item	Size (Bytes)
0	61h	1
1	Page Length = 2	1
2-3	Payload Length = 2	2
4-max	Affected N_Port pages	max

Figure 112. RSCN affected N_Port ID page format

Byte	Item	Size (Bytes)
0	Address Format	1
1-3	Affected N_Port ID	3

The only address format understood by ESS is 00h. All other N_port ID pages are ignored.

Figure 113. RSCN ACC payload

Byte	Item	Size (Bytes)
0-3	ACC Code = 02000000h	4

Report Node Capabilities (RNC)

The report node capabilities (RNC) extended link service request sequence is a single frame providing for the exchange of vendor identification and node capabilities. This link service may be used to query an N_Port to discover what document identifiers it supports. The RNC link service can be used anytime after N_Port Login. Using the RNC link service to query the N_Port's capabilities can be accomplished by setting the RNC_Flags "Select" bit to 0. The requesting N_Port can, but is not required to, supply capability entries in the RNC payload. Upon receiving an RNC request with the "Select" bit set to zero, ESS responds with an RNC accept payload that lists the ESS's capabilities.

Figure 114. RNC payload

Byte	Item	Size (Bytes)
0	Command Code = 53h	1
1	Reserved	1
2-3	Payload Length	2
4	RNC Flags	1
5-6	Reserved	2
7	VU_Len	1
8-15	Vendor Indentifier	8
16-19	Capability Entry	N

Figure 115. RNC ACC payload

Byte	Item	Size (Bytes)	ESS Response
0	ACC Code	1	02h
1	Reserved	1	00h
2-3	Payload Length	2	0018h[8]
4	RNC Flags	1	00h
5-6	Reserved	2	0000h
7	VU_Len	1	00h
8-15	Vendor Indentifier	8	'IBM '
16-19	Capability Entry	4	00047878h[8]
20-23	Capability Entry if FC_AL attached	4	00251B1Bh[8]
20-23	Capability Entry if Point to Point	4	00261515h[8]

Re-instate Recovery Qualifiers (RRQ)

The Re-instate Recovery Qualifier (RRQ) can be sent by the initiator to ESS, as an indication that the Recovery Qualifier (S_ID, D_ID, OX_ID, RX_ID, SEQ_ID, and SEQ_CNT) for an aborted exchange can be reused. RRQ is not required by ESS following an aborted exchange. Following an ABTS, ESS allows immediate reuse of the Recovery Qualifier. ESS responds to RRQ with a valid ACC.

The following tables show the RRQ payload and the RRQ ACC payload.

Figure 116. RRQ payload

Byte	Item	Size (Bytes)
0-3	Command Code = 12000000h	4
4	reserved	1
5-7	Originator S_ID	3
8-9	OX_ID	2
10-11	RX_ID	2
12-43	Association Header (Opt)	32
44-46	N_Port ID of Originator	3

Figure 117. RRQ ACC payload

Byte	Item	Size (Bytes)
0-3	ACC Code = 02000000h	4

ESS performs no protocol checking on the RRQ Payload. When it recognizes an ELS with 12h in byte 0, it simply responds with an RRQ ACC.

---

[8] These are the default RNC ACC payload values. The RNC payload can request a subset of these values in the capability entry (see FC-PH Standards).

State Change Registration (SCR)

The State Change Registration (SCR) Extended Link Service requests the fabric controller to add the N_Port or NL_Port to the list of N_Ports and NL_Ports registered to receive the Registered State Change Notification (RSCN) Extended Link Service.

Figure 118. SCR payload

Byte	Item	Size (Bytes)
0-3	Command Code = 62000000h	4
4-6	reserved	3
7	Registration Function = 3	1

Registration Function is defined as follows:

Value Definition

0 Reserved

1 Fabric Detected registration

Register to receive all RSCN requests issued by the fabric controller for events detected by the fabric.

2 N_Port Detected registration

Register to receive all RSCN requests issued by the fabric controller for events detected by the affected N_Port or NL_Port.

3 Full registration

Register to receive all RSCN requests issued by the fabric controller. The RSCN request shall return all affected N_Port ID pages.

4-254 Reserved

255 Clear registration

Remove any current RSCN registrations.

Third Party Process Logout (TPRLO)

The TPRLO request is used to invalidate the operating environments between the specified images(s) at the recipient N_Port (that is ESS). These image pairs being invalidated are assumed to have been previously established with a Process Login (PRLI). The originator establishing the process image (with the PRLI ELS), can itself terminate the image, with a Process Logout (that is PRLO) ELS. An originator can terminate the process pair established between another third party originator, and the recipient (that is ESS), using TPRLO. TPRLO has the same effect as if the third party originator performed PRLO. Upon execution of a valid TPRLO ELS, ESS will set all tasks, reservations, and status for the specified image pairs to the state they would have after a SCSI hard reset or power on reset.

ESS only supports a single service parameter page. If an attempt to write more than one service parameter page is detected ESS responds with reason code 07h.

Figure 119. TPRLO payload

Byte	Item	Size (Bytes)
0	Command Code = 24h	1
1	Page Length = 14h	1
2-3	Payload Length = 18h	2
4-23	Logout service parameter page	20

Figure 120. TPRLO ACC payload

Byte	Item	Size (Bytes)
0	ACC Cmd Code = 02h	1
1	Page Length = 14h	1
2-3	Payload Length = 18h	2
4-23	Logout service parameter response page	20

Third Party Process Logout Service Parameter Page

Figure 121. Logout Service Parameter page

Byte	Item	Size (Bytes)
0	Type Code = 08h	1
1	Type Code Extension = 00h	1
2-3	Flags	2
4-7	Third Party Originator Process Associator	4
8-11	Responder Process Associator	4
12	Reserved	1
13-15	Third Party Originator N_Port ID	3
16-19	Reserved	4

The Type code field for SCSI-FCP is 08h and the Type Code Extension field is 00h.

The Flags field is as follows:

Bit 15 - Third Party Originator Process Associator Valid
> Must be set to zero marking the Originator Process Associator field of this service parameter page as invalid.

Bit 14 - Responder Process Associator Valid
> Must be set to zero marking the Responder Process Associator field of this service parameter page as invalid.

Bit 13 - Third Party Originator N_Port ID Valid
> When set (1), the Third Party N_Port ID field of this service parameter page is valid.

Bit 12 - Global Process Logout
> When set (1), all established image pairs for all N_Ports with which Process Login has been performed are removed. This is as if all N_Ports that had previously executed a Process Login (that is PRLI) with ESS, had now executed the PRLO ELS. When this bit is set, only

one logout parameter page shall be transmitted, and only the Type Code and the Type Code Extension fields shall have meaning.

Bits 11-0 - Reserved

The Third Party Originator Process Associator identifies a group of related processes (an image) within the originator.

The Responder Process Associator identifies a group of related processes (an image) within the responder.

The Third Party N_Port ID specifies the N_Port associated with the image to be removed.

Third Party Process Logout Response Service Parameter Page

Figure 122. Logout Response Service Parameter page

Byte	Item	Size (Bytes)	ESS Response
0	Type Code	1	08
1	Type Code Extension	1	00
2-3	Flags	2	XX 00
4-7	Third Party Originator Process Associator	4	00 00 00 00
8-11	Responder Process Associator	4	00 00 00 00
12	Reserved	1	00
13-15	Third Party Originator N_Port ID	3	XX XX XX
16-19	Reserved	4	00 00 00 00 00

The Type code field for SCSI-FCP is 08h and the Type Code Extension field is 00h.

The Flags field is as follows:

Bit 15 - Third Party Originator Process Associator Valid
    ESS returns 0b in this field.

Bit 14 - Responder Process Associator Valid
    ESS returns 0b in this field.

Bit 13 - Third Party Originator N_Port ID Valid
    ESS echos this bit from the TPRLO payload.

Bit 12 - Global Process Logout
    ESS echos this bit from the TPRLO payload.

Bits 11-8 - Accept Response Code
    See Figure 101 on page 109.

Bits 7-0 - Reserved

The Originator Process Associator and Responder Process Associator fields are not used.

ESS echos the Third Party N_Port ID field from the TPRLO payload.

FC Timers

Figure 123. FCAL timer values

Timer	Description	Value
AL_TIME	Arbitrated Loop Timeout Value	15 ms
LIS_HOLD_TIME	Loop Initialization Sequence Hold Time	1 ms
R_T_TOV	Receiver Transmitter Timeout Value	100 ms
E_D_TOV	Error Detect Timeout Value	2 sec
R_A_TOV	Resource Allocation Timeout Value	2 sec
RR_TOV	Resource Recovery Timeout Value	2 sec
ULP_TOV	Upper Level Protocol Timeout Value	10 sec

The timers are used by ESS as required.

AL_TIME AL_TIME represents two times the worst case round trip latency for a very large loop.

LIS_HOLD_TIME LIS_HOLD_TIME is the maximum amount of time between when a node receives a loop initialization sequence until it forwards it to the next node.

R_T_TOV R_T_TOV is used by the receiver logic to detect loss of synchronization.

E_D_TOV E_D_TOV is a general error detect timeout value. It is used to time the following:

- The maximum time permitted for a sequence initiator between the transmission of consecutive data frames within a single sequence.

- The minimum time that a sequence recipient shall wait for the reception of the next frame within a single sequence before recognizing a sequence timeout.

- The minimum time a private NL_Port shall wait to win arbitration before it can assume a malfunction or excessive unfairness and optionally transmit a LIP sequence.

E_D_TOV shall include the time required to gain access to the loop in addition to the actual frame transmission time.

R_A_TOV A value of twice R_A_TOV is used to determine the minimum time that ESS will wait for the response to a Link Service or FC-4 Extended Link Service request.

RR_TOV RR_TOV is the minimum time a target shall wait for an initiator to perform exchange authentication following LIP. If this timer expires, the target implicitly logs out the initiator and frees up the resources associated with that timer. An RR_TOV timer is maintained for each initiator that has logged in.

ULP_TOV This is an operation-specific timer maintained by the upper level protocol. The minimum value used for ULP_TOV shall be E_D_TOV. It is used to time the completion of exchanges associated with ULP operations. Because the amount of time required varies depending upon the operation, the value assigned for this timer is determined by the operation being timed.

Link Failure Link Failure is defined when a receiver has continuously detected loss of synchronization for a period of R_T_TOV. When a link failure condition is detected on a port, it generates NOS if in point-to-point, or transmit LIPF8 if Arbitrated Loop.

Register FC-4 Type (RFC-4)

As a part of initialization on an FC port, ESS attempts to register its FC-4 type with the name server if it is able to complete a FLOGI on a port that is configured as point-to-point.

Registration is done using a CT_IU as specified in *FC_GS-2 rev 5.0*:

- Sent to the well known address X'FFFFFC'

- Sent class 2

- A type X'20'

- Routing/IU cat 'X'02'

The IU payload sent is:

Figure 124. Register FC-4 Type payload

Byte	Item	Size (Bytes)	ESS Value
0	Revision	1	1
1	IN_ID	3	0
4	FCS_Type	1	X'FC'
5	FCS_Subtype	1	2
6	Options	1	0
7	Reserved	1	0
8	Command / Response Code	2	X'0217'
10	Maximum Residual Size Field	2	12
12	Reserved	1	0
13	Reason Code	1	0
14	Reason Code Explanation	1	0
15	Vendor Unique	1	0
16	Reserved	1	0
17	Port Name	3	Registering port's fabric assigned S_ID
20	FC-4 Types	32	X'18000100' followed by 28 bytes of zeros

FCP

| **FCP header**

| The format of the frame header used by the FCP is defined in Figure 125.

| *Figure 125. FCP header*

Word	Byte 0	Byte 1	Byte 2	Byte 3
0	R_CTL		D_ID	
1	reserved		S_ID	
2	Type		F_CTL	
3	SEQ_ID	DF_CTL	SEQ_CNT	
4	OX_ID		RX_ID	
5	RLTV_OFF			

| **R_CTL**
| The R_CTL field identifies the frame as part of an FCP operation and identifies the
| information category. All sequences (IUs) are of Device_Data type frame and the
| information category associated with each IU is defined in Figure 127 and
| Figure 128.

| **D_ID**
| The D_ID identifies the destination of the frame. In ESS, D_ID is the destination
| node ID.

| **S_ID**
| The S_ID identifies the source of the frame. In ESS, S_ID is the source node ID.

| **Type**
| The Type field value is 0x08 for all frames of SCSI FCP sequences.

| **F_CTL**
| The F_CTL field, shown in Figure 126, contains the control information relating to
| the frame content.

| **SEQ_ID**
| The SEQ_ID uniquely identifies each sequence within an exchange.

| **DF_CTL**
| The DF_CTL indicates any optional headers that may be present. This is **not** used
| by ESS.

| **SEQ_CNT**
| The SEQ_CNT indicates the frame order within the sequence.

| **OX_ID**
| The OX_ID is the originator identification of the exchange. The initiator assigns a
| unique value other than 0xFFFF for OX_ID in the first sequence of an exchange.
| The value of OX_ID is the tag value defined by SCSI.

RX_ID

The RX_ID is the responder identification of the exchange. If RX_ID is being used, the target assigns a unique value other than 0xFFFF in the first sequence transmitted as a sequence initiator, if any. An RX_ID of 0xFFFF indicates that the RX_ID is unassigned.

RLTV_OFF

The RLTV_OFF specifies the relative offset of the first byte of each frame's payload or data with reference to the base address of the information category.

Figure 126. F_CTL field

Bits	Control Field	Description	ESS Usage
23	Exchange Context	Originator (0) or Responder (1) of exchange	yes
22	Sequence Context	Sequence initiator (0) or Sequence recipient (1)	yes
21	First_Sequence	First sequence of exchange	yes
20	Last_Sequence	Last sequence of exchange	yes
19	End_Sequence	Last data frame of sequence	yes
18	End_Connection	Only meaningful in Class 1 (Dedicated connection)	no
17	Chained Sequence	Only meaningful in Class 1 (Dedicated connection)	no
16	Sequence Initiative	Hold (0) or Transfer (1) sequence initiative	yes
15	X_ID reassigned	Only meaningful if reassigning X_ID	no
14	Invalidate X_ID	Only meaningful if reassigning X_ID	no
13-10	reserved		
9	Retransmitted Sequence	Only meaningful in Class 1 (Dedicated connection)	no
8	Unidirectional Transmit	Only meaningful in Class 1 (Dedicated connection)	no
7-6	Continue Sequence Condition	Estimated transmission time for the next sequence of the current exchange	no
5-4	Abort Sequence Condition		no
3	Relative Offset Present	RLTV_OFF is valid in FCP header	yes
2	Exchange Reassembly	Reserved for exchange reassembly	no
1-0	Fill Data Bytes	End of data field - bytes of fill	no

List of IUs supported

The information units supported by ESS are shown in Figure 127 for IUs sent to targets, and in Figure 128 for IUs sent to initiators.

Figure 127. FCP IUs sent to targets

IU	SCSI-3 Primitive	Data Block		F/M/L	SI	M/O
		CAT	Content			
T1	Command/Task Mgmt request	6	FCP_CMND	F	T	M
T2	Command/Task Mgmt request	6	FCP_CMND	F	H	O
T3	Command request (linked)	6	FCP_CMND	M	T	O
T4	Command request (linked)	6	FCP_CMND	M	H	O
T6	Data Out action	1	FCP_DATA	M	T	M

Key:

IU Information Unit
CAT Information Category of Device_Data frame
CONTENT Contents (payload) of data block
F/M/L First/Middle/Last Sequence of exchange
SI Sequence Initiative: Held or Transferred
M/O Mandatory/Optional Sequence

Figure 128. FCP IUs sent to initiators

IU	SCSI-3 Primitive	Data Block		F/M/L	SI	M/O
		CAT	Content			
I1	Data delivery request	5	FCP_XFER_RDY (Write Only)	M	T	M
I3	Data in action	1	FCP_DATA	M	H	M
I4	Command/Task Mgmt Response	7	FCP_RSP	L	T	M
I5	Command Response (linked)	7	FCP_RSP	M	T	O

Key:

IU Information Unit
CAT Information Category of Device_Data frame
CONTENT Contents (payload) of data block
F/M/L First/Middle/Last Sequence of exchange
SI Sequence Initiative: Held or Transferred
M/O Mandatory/Optional Sequence

FCP_CMND IU

The FCP_CMND IU carries either a SCSI command to be executed or a task management function to be performed. The payload of the FCP_CMND IU is shown in Figure 129.

Figure 129. FCP_CMND Payload

Byte	Field	Description	Size (Bytes)
0-7	LUN_ID	Logical Unit Number	8
8-11	FCP_CNTL	Control Field	4
12-27	FCP_CDB	SCSI command descriptor block	16
28-31	FCP_DL	Data Length	4

LUN_ID

The 8-byte LUN_ID field is used to address the desired logical unit. The 8-byte LUN structure allows up to four levels of device addressing as described in the *SCSI-3 Controller Commands (SCC)* standard. ESS only uses the first level addressing in bytes 0 and 1 as defined in Figure 130. ESS uses either the virtual device ('01'b) or the peripheral device ('00'b) addressing method depending on whether the initiator is configured as supporting the Report LUNs command.

Figure 130. LUN_ID

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
For hosts, configured as using Report LUNs:								
0	Addr_Method '01'b (Virtual Device)		LUN					
1	LUN							
2-7	0							
For hosts, configured not as using Report LUNs:								
0	Addr_Method '00'b (Peripheral Device)		BUS = 0					
1	LUN							
2-7	0							

FCP_CNTL

The FCP_CNTL field contains a number of control flags as shown in Figure 131.

Figure 131. FCP_CNTL field format

Byte	Description	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	reserved	reserved							
1	Task Codes	reserved				TASK ATTRIBUTE			
2	Task Management Flags	obsolete	CLEAR ACA	TARGET RESET	LUN RESET	rsvd	CLEAR TASK SET	ABORT TASK SET	rsvd
3	Execution Management Codes	reserved						READ DATA	WRITE DATA

Task Codes

The Task Attribute values are shown in Figure 132 on page 125. The task with the specified Task Attribute is treated according to the Tagged Queuing, Untagged

Queuing, and Auto Contingent Allegiance rules as described in the *Seascape SCSI-3 DASD Initial Attachment Specification*.

Figure 132. Task attribute values

Value	Task Attribute
000b	SIMPLE_Q
001b	HEAD_OF_Q
010b	ORDERED_Q
100b	ACA_Q
101b	UNTAGGED
Others	RESERVED

Task management flags

Task management flags are used to request the task management function to be performed. Terminate Task is not being used by ESS. Any other task management function is transmitted using a new exchange. If any task management flag is set, all CDB related fields (FCP_CDB, FCP_DL, task codes and execution management codes of FCP_CNTL) are ignored. Only one task management flag can be set in any FCP_CMND IU.

A task management function ends with an FCP_RSP IU that indicates whether it was correctly accepted.

Terminate Task The Terminate Task task management flag is not supported by ESS.

Clear ACA The CLEAR ACA task management flag causes ESS to execute the CLEAR ACA task management function. ESS always returns an FCP_RSP IU with a RSP_CODE of `No Failure/Task Management function complete`, even if ACA was not cleared.

Target Reset The Target Reset task management flag causes ESS to execute the Target Reset task management function.

Clear Task Set The Clear Task Set task management flag causes ESS to perform the Clear Task Set task management function.

Abort Task Set The abort task set task management flag causes ESS to perform the Abort Task Set task management function.

Execution management codes

Read Data is a SCSI read operation. It indicates that the initiator expects the FCP_DATA IU to be in the opposite direction of the FCP_CMND IU.

Write Data is a SCSI write operation. It indicates that the initiator expects the FCP_DATA IU to be in the same direction as the FCP_CMND IU.

If both Read Data and Write Data are not set, then there should be no FCP_DATA IUs for this exchange, and also FCP_DL should be 0. Only one of these two bits can be set for each FCP_CMND IU.

FCP_CDB

The FCP_CDB is a 16-byte field that contains the SCSI CDB to be executed by the addressed logical unit. Bytes beyond the last byte of the SCSI CDB are ignored by the target and may have any value.

This field is ignored if any of the task management flags is set.

FCP_DL

The FCP_DL field contains a count of the number of data bytes expected to be transferred by the SCSI CDB. An FCP_DL value of 0 indicates that no data transfer is expected and that no FCP_XFER_RDY or FCP_DATA IUs shall be transferred.

FCP_XFER_RDY IU

The FCP_XFER_RDY IU indicates that the target is prepared to perform all or part of the data transfer for a command.

During write operations, this IU indicates the amount of data the target expects from the initiator. Since the target has prepared the buffer resources based on that amount of data, the initiator should provide the exact amount requested. The target sends this IU to request each FCP_DATA IU from the initiator.

ESS does **not** use this IU during read operations.

The payload of the FCP_XFER_RDY IU is shown in Figure 133.

Figure 133. FCP_XFER_RDY payload

Byte	Field	Description	Size (Bytes)
0-3	DATA_RO	Relative Offset of first byte of FCP_DATA IU that follows	4
4-7	BURST_LEN	Length of FCP_DATA IU that follows	4
8-11	reserved		4

DATA_RO

The DATA_RO field indicates the contents of the RLTV_OFF field for the first data byte of the next FCP_DATA IU. The RLTV_OFF field is part of the FCP header and is described in "FCP header" on page 121.

BURST_LEN

The BURST_LEN field indicates the amount of buffer space the target has prepared for the next FCP_DATA IU and requests the transfer of an IU of that exact length. A BURST_LEN of 0 is not valid.

FCP_DATA IU

During write operations, each FCP_DATA IU is preceded by an FCP_XFER_RDY IU. During read operations, the FCP_DATA IU is transmitted without a preceding FCP_XFER_RDY IU.

The target should never request or deliver data exceeding the data length defined by FCP_DL.

| FCP_RSP IU

| The content of the FCP_RSP IU is shown in Figure 134.

| *Figure 134. FCP_RSP payload*

Byte	Field	Description	Size (Bytes)
0-7	reserved		8
8-11	FCP_STATUS	Field Validity and SCSI Status	4
12-15	FCP_RESID	Residual Count	4
16-19	FCP_SNS_LEN	Length of FCP_SNS_INFO field (0 if no sense data, 32 if sense data)	4
20-23	FCP_RSP_LEN	Length of FCP_RSP_INFO field (0 if no RSP info, 8 if RSP info)	4
24->	FCP_RSP_INFO	FCP Response Information	m
24+m->	FCP_SNS_INFO	SCSI Sense Information	n

| FCP_STATUS

| For unlinked commands, the FCP_STATUS field is normally 0 upon successful completion of an I/O operation. For linked commands, Intermediate status indicates successful completion of a command.

| *Figure 135. FCP_STATUS field format*

Byte	Description	Size (Bytes)
0-1	reserved	2
2	Field validity flags Bits 7-4 reserved Bit 3 RESID_UNDER Bit 2 RESID_OVER Bit 1 SNS_LEN_VALID Bit 0 RSP_LEN_VALID	1
3	SCSI status byte	1

| The reserved bytes 0 and 1 are set to zero.

| Byte 2 contains the validity and status indicators for the subsequent fields.

| RESID_UNDER indicates that the FCP_RESID field is valid and contains the count of bytes expected to be transferred, but that were not transferred.

| RESID_OVER indicates that the FCP_RESID field is valid and contains the count of bytes that could not be transferred because the FCP_DL was not sufficient.

| SNS_LEN_VALID indicates that the FCP_SNS_LEN field is valid and contains the count of bytes of the FCP_SNS_INFO field.

| RSP_LEN_VALID indicates that the FCP_RSP_LEN field is valid and contains the count of bytes of the FCP_RSP_INFO field.

| Byte 3 contains the actual SCSI status codes.

FCP_RESID

If the RESID_UNDER or the RESID_OVER bit is 1, the FCP_RESID field contains a count of the residual data bytes that were not transferred for this SCSI command.

If the RESID_UNDER or the RESID_OVER bit is 0, the FCP_RESID field is not valid and can contain any value.

FCP_SNS_LEN

If the SNS_LEN_VALID bit is 1, the FCP_SNS_LEN field contains a count of the valid bytes in FCP_SNS_INFO. If SNS_LEN_VALID bit is 0, this field is not valid and no FCP_SNS_INFO is provided.

FCP_RSP_LEN

If the RSP_LEN_VALID bit is 1, the FCP_RSP_LEN field contains a count of the valid bytes in FCP_RSP_INFO. Valid count values are 0, 4 and 8. If RSP_LEN_VALID bit is 0, this field is not valid and no FCP_RSP_INFO is provided.

FCP_RSP_INFO

The FCP_RSP_INFO field contains information describing only the protocol failures detected during the execution of an I/O operation. This field does not contain link error or SCSI logical unit error information.

Figure 136. FCP_RSP_INFO field format

Byte	Description	Size (Bytes)
0-2	reserved, set to zero.	3
3	RSP_CODE	1
4-7	reserved	4

The valid RSP_CODE values are defined in Figure 137.

Figure 137. RSP_CODE definitions

Code	Definition
00h	No Failure / Task Management function complete
01h	FCP_DATA length different than BURST_LEN
02h	FCP_CMND Fields invalid
03h	FCP_DATA RO mismatch with FCP_XFER_RDY DATA_RO
04h	Task Management Function not supported
05h	Task Management Function failed
06h-FFh	reserved

FCP_SNS_INFO

The FCP_SNS_INFO field contains the sense information specified by SCSI as described in the *Seascape SCSI-3 DASD Initial Attachment Specification*. The FCP_SNS_INFO is presented only if the SCSI status byte in the FCP_STATUS field indicates a Check Condition status.

Parallel SCSI-3 to FCP-SCSI message conversion

Figure 138 defines the mapping (if any) between the functions of the SCSI-3 messages and FCP-SCSI.

Figure 138. Mapping from parallel SCSI-3 messages to FCP-SCSI

SCSI-3 Message	FCP Equivalent
Abort Task	ABTS
Abort Task Set	FCP_CMND (FCP_CNTL ABORT TASK SET)
ACA	FCP_CMND (FCP_CNTL TASK ATTRIBUTE ACA_Q)
Clear ACA	FCP_CMND (FCP_CNTL CLEAR ACA)
Clear Task Set	FCP_CMND (FCP_CNTL CLEAR TASK SET)
Disconnect	N/A
Head of Queue Tag	FCP_CMND (FCP_CNTL TASK ATTRIBUTE HEAD_OF_Q)
Identify (In)	N/A
Identify (Out)	FCP_CMND (FCP_LUN)
Ignore Wide Residue	N/A
Initiator Detected Error	N/A
Linked Command Complete	FCP_RSP (FCP_STATUS)
LUN Reset	FCP_CMND (FCP_CNTL LUN RESET)
Message Parity Error	N/A
Message Reject	FCP_RSP (FCP_RSP_INFO)
No Operation	N/A
Ordered Queue Tag	FCP_CMND (FCP_CNTL TASK ATTRIBUTE ORDERED_Q)
Simple Queue Tag	FCP_CMND (FCP_CNTL TASK ATTRIBUTE SIMPLE_Q)
Save Data Pointer	N/A
Synchronous Data Transfer Request	N/A
Target Reset	FCP_CMND (FCP_CNTL TARGET RESET)
Task Complete	FCP_RSP (FCP_STATUS)
Wide Data Transfer Request	N/A

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

> IBM Director of Licensing
> IBM Corporation
> North Castle Drive
> Armonk, NY 10504-1785
> U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publications. IBM may make improvements and/or changes in the product(s) and/or program(s) described in this publication at any time without notice.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Read Before Using

IMPORTANT

YOU ACCEPT THE TERMS OF THIS IBM LICENSE AGREEMENT FOR MACHINE CODE BY YOUR USE OF THE HARDWARE PRODUCT OR MACHINE CODE. PLEASE READ THE AGREEMENT CONTAINED IN THIS BOOK BEFORE USING THE HARDWARE PRODUCT. SEE "IBM Agreement for Licensed Internal Code" on page 135.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

AIX
Application System/400
AS/400
Enterprise Storage Server
ESCON
IBM
MVS
Netfinity
OS/390
OS/400
RISC System/6000
RS/6000
RISC System/6000 Series Parallel
RS/6000 SP
S/370
S/390
Seascape
StorWatch
System/390

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

Other company, product, and service names may be trademarks or servic e marks of others.

Electronic emission notices

This section gives the electronic emission notices or statements for the United States and other countries.

Federal Communications Commission (FCC) statement

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

Properly shielded and grounded cables and connectors must be used in order to meet FCC emission limits. IBM is not responsible for any radio or television interference caused by using other than recommended cables and connectors or by unauthorized changes or modifications to this equipment. Unauthorized changes or modifications could void the user's authority to operate the equipment.

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

Industry Canada compliance statement

This Class A digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de la classe A est conform à la norme NMB-003 du Canada.

European Community compliance statement

This product is in conformity with the protection requirements of EC Council Directive 89/336/EEC on the approximation of the laws of the Member States relating to electromagnetic compatibility. IBM cannot accept responsibility for any failure to satisfy the protection requirements resulting from a non-recommended modification of the product, including the fitting of non-IBM option cards.

Germany only

Zulassungsbescheinigung laut Gesetz ueber die elektromagnetische Vertraeglichkeit von Geraeten (EMVG) vom 30. August 1995.

Dieses Geraet ist berechtigt, in Uebereinstimmung mit dem deutschen EMVG das EG-Konformitaetszeichen - CE - zu fuehren.

Der Aussteller der Konformitaetserklaeung ist die IBM Deutschland.

Informationen in Hinsicht EMVG Paragraph 3 Abs. (2) 2:

```
Das Geraet erfuellt die Schutzanforderungen nach EN 50082-1 und
  EN 55022 Klasse A.
```

EN 55022 Klasse A Geraete beduerfen folgender Hinweise:

Nach dem EMVG:

```
"Geraete duerfen an Orten, fuer die sie
nicht ausreichend entstoert
sind, nur mit besonderer Genehmigung des Bundesministeriums
fuer Post und Telekommunikation oder des Bundesamtes fuer Post und
Telekommunikation
betrieben werden. Die Genehmigung wird erteilt, wenn keine
elektromagnetischen Stoerungen zu erwarten sind." (Auszug aus dem
EMVG, Paragraph 3, Abs.4)
```

```
Dieses Genehmigungsverfahren ist nach Paragraph 9 EMVG in Verbindung
mit der entsprechenden Kostenverordnung (Amtsblatt 14/93)
kostenpflichtig.
```

Nach der EN 55022:

```
"Dies ist eine Einrichtung der Klasse A. Diese Einrichtung kann im
Wohnbereich Funkstoerungen verursachen. in diesem Fall kann vom
Betreiber verlangt werden, angemessene Massnahmen durchzufuehren
und dafuer aufzukommen."
```

Anmerkung:

```
Um die Einhaltung des EMVG sicherzustellen, sind die Geraete wie in den
Handbuechern angegeben zu installieren und zu betreiben.
```

Japanese Voluntary Control Council for Interference (VCCI) Class A statement

この装置は、情報処理装置等電波障害自主規制協議会（ＶＣＣＩ）の基準
に基づくクラスＡ情報技術装置です。この装置を家庭環境で使用すると電波
妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ず
るよう要求されることがあります。

VS1KAN

Korean Government Ministry of Communication (MOC) statement

Please note that this device has been approved for business purpose with regard to electromagnetic interference. If you find this is not suitable for your use, you may exchange it for a non-business purpose one.

Taiwan Class A Compliance statement

警告使用者:
這是甲類的資訊產品，在居住的環境中使用
時，可能會造成射頻干擾，在這種情況下，
使用者會被要求採取某些適當的對策。

IBM Agreement for Licensed Internal Code

You accept the terms of this Agreement[9] by your initial use of a machine that contains IBM Licensed Internal Code (called "Code"). These terms apply to Code used by certain machines IBM or your reseller specifies (called "Specific Machines"). International Business Machines Corporation or one of its subsidiaries ("IBM") owns copyrights in Code or has the right to license Code. IBM or a third party owns all copies of Code, including all copies made from them.

If you are the rightful possessor of a Specific Machine, IBM grants you a license to use the Code (or any replacement IBM provides) on, or in conjunction with, only the Specific Machine for which the Code is provided. IBM licenses the Code to only one rightful possessor at a time.

Under each license, IBM authorizes you to do only the following:

1. Execute the Code to enable the Specific Machine to function according to its Official Published Specifications (called "Specifications");

2. Make a backup or archival copy of the Code (unless IBM makes one available for your use), provided you reproduce the copyright notice and any other legend of ownership on the copy. You may use the copy only to replace the original, when necessary; and

3. Execute and display the Code as necessary to maintain the Specific Machine.

You agree to acquire any replacement for, or additional copy of, Code directly from IBM in accordance with IBM's standard policies and practices. You also agree to use that Code under these terms.

You may transfer possession of the Code to another party only with the transfer of the Specific Machine. If you do so, you must 1) destroy all your copies of the Code that were not provided by IBM, 2) either give the other party all your IBM-provided copies of the Code or destroy them, and 3) notify the other party of these terms. IBM licenses the other party when it accepts these terms. These terms apply to all Code you acquire from any source.

Your license terminates when you no longer rightfully possess the Specific Machine.

Actions You Must Not Take

You agree to use the Code only as authorized above. You must not do, for example, any of the following:

1. Otherwise copy, display, transfer, adapt, modify, or distribute the Code (electronically or otherwise), except as IBM may authorize in the Specific Machine's Specifications or in writing to you;

2. Reverse assemble, reverse compile, or otherwise translate the Code unless expressly permitted by applicable law without the possibility of contractual waiver;

3. Sublicense or assign the license for the Code; or

4. Lease the Code or any copy of it.

---

9 Form Z125-4144.

Glossary

A

| **ADISC**. address discovery

allegiance. The ESA/390 term for a relationship that is created between a device and one or more channel paths during the processing of certain conditions. See also implicit allegiance, contingent allegiance, and reserved allegiance.

ANSI. American National Standards Institute. A standards committee body for the United States.

ASCII. A coding scheme defined by ANSI X3.4-1977 which is used to represent various alphabetic, numeric, and special symbols with a seven-bit code.

B

bit. binary digit. The storage medium required to store a single binary digit.

block. A group of consecutive bytes.

byte. An aggregation of eight bits. The storage medium required to store eight bits.

C

cache memory. Typically refers to volatile memory used to improve access times on data or instructions accesses. The cache memory is typically smaller and faster than the primary memory or storage medium. On a storage facility, the cache memory is used to improve the performance of accesses to data that ultimately resides on the storage devices in the storage facility.

CCR. channel command retry.

CCW. channel command word. The ESA/390 term for a data structure that is used to specify an I/O operation to the channel subsystem.

CEC. computer electronic complex. The set of hardware facilities associated with a host computer.

channel. The ESA/390 term for the part of a channel subsystem that manages a single I/O interface between a channel subsystem and a set of control units.

channel-command retry. An ESA/390 term for a protocol used between a channel and a control unit which allows the control unit to request that the current command be reissued.

channel path. The ESA/390 term for the interconnection between a channel and its associated control units.

channel subsystem. The ESA/390 term for the part of a host computer that manages I/O communication between the program and any attached control units.

channel-subsystem image. The ESA/390 term for the logical facilities required to perform the function of a channel subsystem. With EMIF, one channel-subsystem image exists in the channel subsystem for each LPAR. Each image appears to be an independent channel subsystem program, but all images share a common set of hardware facilities.

CKD. count key data. An ESA/390 architecture for a DASD logical device that specifies the format of and access mechanisms for the logical data units on the device. The logical data unit is a track which can contain one or more records, each consisting of a count field, a key field (optional), and a data field (optional).

cluster. A partition of a storage server which is capable of performing all defined functions of the storage server. Multiple clusters provide redundancy.

Concurrent Copy. A facility of a storage server used to assist a program in making a consistent copy of a logical volume while the logical volume remains available for subsequent processing.

consistent copy. A copy of a data entity (e.g. a logical volume) that contains the contents of the entire data entity from a single instant in time.

contingent allegiance. The ESA/390 term for a relationship that is created in a control unit between a device and a channel path when unit-check status is accepted by the channel. The allegiance causes the control unit to guarantee access (i.e. busy status is not presented) to the device to retrieve the sense data associated with the unit-check status on the channel path associated with the allegiance.

control unit. The ESA/390 term for a storage server with ESCON or OEMI interfaces.

control-unit image. The ESA/390 term for a logical subsystem accessed via an ESCON I/O Interface. One or more control-unit images exist in each control unit. Each image appears to be an independent control unit, but all images share a common set of hardware facilities.

count field. The first field of a CKD record. The eight-byte field contains a four-byte track address

(CCHH) that defines the cylinder and head associated with the track, a one-byte record number (R) that identifies the record on the track, a one-byte key length that specifies the length of the record's key field (0 means no key field), and a two-byte data length that specifies the length of the record's data field (0 means no data field). Only the end-of-file record has a data length of zero.

CPAR.   customer problem analysis and repair.

CPSS.   Common Platform Storage Subsystems. A strategy and family of storage products based on common hardware parts.

CRC.   cyclic redundancy check.

CSU.   customer set up.

cylinder.   A unit of storage on a CKD device consisting of a fixed number of tracks.

D

DA.   device adapter.

DASD.   Direct Access Storage Device. A disk device.

DASD Fast Write.   A facility of a storage server that allows caching of active write data without exposure of data loss by journaling of the active write data in NVS.

data compression.   A technique or algorithm used to encode a quantity of data such that the encoded result can generally be stored in less space than the original data and such that the original data can be recovered from the encoded result through a reverse technique or reverse algorithm.

data field.   The third (optional) field of a CKD record. The field length is determined by the data length specified in the count field. The data field contains data written by the program.

data record.   A data records are stored on a track following the track-descriptor record. The data records are numbered consecutively starting with 1. A maximum of 255 data records can be stored on a track. Each data record consists of a count field, a key field (optional), and a data field (optional).

demote.   The action of removing a logical data unit from cache memory. A data unit is demoted in order to make room for other logical data units in the cache or because the logical data unit was invalidated. Logical data units with active write unit must be destaged before being demoted.

destage.   Destage is the process of moving data that has been modified in cache to the DDMs.

device.   The ESA/390 term for a logical device.

device adapter.   A physical sub-unit of a storage server that provides the facilities to attach to one or more interfaces used to communicate with the associated storage devices.

device address.   The ESA/390 term for the field of an ESCON device-level frame that selects a specific device on a control-unit image.

device number.   The ESA/390 term for a four-hexadecimal-character identifier (e.g. X'13A0') associated with a device to facilitate communication between the program and the host operator. The device number is associated with a subchannel.

E

EBCDIC.   A coding scheme developed by IBM which is used to represent various alphabetic, numeric, and special symbols with an 8 bit code.

EC.   engineering change. A update to a machine, part, or program. Each EC for a given unit is assigned a unique number refered to as an EC level or EC number. The initial release of a unit normally has an associated EC level.

ECC.   error correction code.

ECKD.   extended count key data. An extension of the CKD architecture.

EMIF.   ESCON Multiple Image Facility. An ESA/390 facility that allows LPARs to share an ESCON channel path by providing each LPAR with its own channel-subsystem image.

end of file.   On a CKD DASD device, end of file is indicated by a record with a data length of zero.

ERP.   error recovery procedure.

ESA/390.   Enterprise Systems Architecture/390 An IBM architecture for mainframe computers and peripherals. Processor systems following this architecture include the ES/9000 family.

ESCON.   Enterprise Systems Connection Architecture. An ESA/390 computer peripheral interface. The I/O interface utilizes ESA/390 logical protocols over a serial interface that configures attached units to a communication fabric.

ESCON director.   A I/O interface switch that allows the interconnection of multiple ESCON interfaces in a distributed-star topology.

Extended Remote Copy. A facility of a storage server that assists a control program to maintain a consistent copy of a logical volume on another storage facility. All modifications of the primary logical volume by any attached host are presented in order to a single host which in turn makes these modifications on the secondary logical volume.

F

FBA. fixed block address. An architecture for DASD logical devices that specifies the format of and access mechanisms for the logical data units on the device. The logical data unit is a block. All blocks on the device are the same size (i.e. fixed) and can be accessed independently,

FC-AL. Fibre Channel - Arbitrated Loop. An implementation of the fibre channel standard that uses a ring topology for the communication fabric. Reference ANSI X3T11/93-275.

FCS. Fibre Channel Standard.

fibre channel standard. An ANSI standard for a computer peripheral interface. The I/O interface defines a protocol for communication over a serial interface that configures attached units to a communication fabric. The protocol is defined in two layers. The IP layer defines basic interconnection protocols. The upper layer supports one or more logical protocols (e.g. FCP for SCSI command protocols, SBCON for ESA/390 command protocols). Reference ANSI X3.230-199x.

FIFO. first-in first-out.

| **FLOGI**. fabric login

FRU. field-replaceable unit.

H

HA. home address, host adapter.

HDA. head and disk assembly. The portion of a HDD associated with the medium and the read/write head.

HDD. head and disk device. A disk drive.

home address. A nine-byte field at the beginning of a track that contains information identifying the physical track and its association with a cylinder.

host. A computer.

host adapter. A physical sub-unit of a storage server that provides the facilities to attach to one or more host I/O interfaces.

I

I/O. input/output.

I/O interface. An interface defined to allow input/output operations to be performed between a host and its associated peripheral devices.

IML. initial microcode load

implicit allegiance. The ESA/390 term for a relationship that is created in a control unit between a device and a channel path, or path group, when an I/O operation is accepted by the device. The allegiance causes the control unit to guarantee access (i.e. busy status is not presented) to the device for the remainder of the channel program over the set of channel paths associated with the allegiance.

initiator. A SCSI term for the part of a host computer that communicates with its attached targets.

interchange. The capability of processing (read or write) a given medium on any one of a set of devices which support the form factor and recording format on the medium.

invalidate. The action of removing a logical data unit from cache memory because of an inability to support continued access to the logical data unit on the logical device. The inability may be the result of a failure within the storage server or a storage device associated with the logical device.

IPL. initial program load

K

key field. The second (optional) field of a CKD record. The field length is determined by the key length specified in the count field. The key field contains data written by the program that can be used to identify or locate a given record.

KPOH. thousands of power-on hours. A unit of time used for a measurement of MTBF.

L

LIC. licensed internal code

licensed internal code. Firmware, microcode, or software installed on a product that is required by the product to perform its defined function. Licensed internal code is provided by the product vendor with the product.

LIFO. last-in first-out

link address. On an ESCON interface, the portion of a source or destination address in a frame used route a frame through an ESCON director. The link address is associated with a specific switch port on the ESCON director, or equivalently, with the channel-subsystem or control-unit link-level facility attached to that switch port.

link-level facility. An ESCON term for the hardware and logical facilities of a control unit or channel subsystem required to allow communication over an ESCON I/O interface.

logical address. On an ESCON interface, the portion of a source or destination address in a frame used to select a specific channel-subsystem or control-unit image.

logical data unit. A unit of storage which is accessible on a given logical device.

logical device. The facilities of a logical subsystem that the host communicates with when performing I/O operations to a single addressable-unit over an I/O interface. The same logical device may be accessible over more than one I/O interface.

logical subsystem. The logical facilities in a storage server required to allow one or more host I/O interfaces to access a set of logical devices that are considered to be aggregated according to the addressing mechanisms of the associated I/O interfaces. One or more logical subsystems exist on a storage server. In general, a given set of logical devices is associated with one and only one logical subsystem.

logical unit. The SCSI term for a logical device.

logical unit number. The SCSI term for the field in an Identify message that is used to select a logical unit on a given target.

logical volume. The storage medium associated with a logical device. A logical volume typically resides on one or more storage devices.

LPAR. logical partition. The ESA/390 term for a set of facilities that create the programming environment defined by the ESA/390 architecture. The term is typically used when there is more than one LPAR established on a CEC. An LPAR is conceptually similar to a virtual machine environment except that the LPAR is a function of the CEC and is not dependent on an operating system to create the virtual machine environment.

LRU. least recently used. A policy for a caching algorithm which chooses to remove the item from cache which has the longest elapsed time since its last access.

LUN. logical unit number.

M

MLC. machine-level control. A data bases that maintains the EC level and configuration of products in the field.

MTBF. mean time between failures. A projection of the time that an individual unit will remain functional based on averaging the performance or projected performance of a population of statistically independent units under as set of operating conditions or assumptions.

N

ND. node descriptor.

NED. node-element descriptor.

NEQ. node-element qualifier.

non-removable medium. Recording media that cannot be added to or removed from a storage device.

non-retentive data. Data that has the attribute that it can easily recreated by the control program in the event it is lost. Non-retentive write data may be cached in volatile memory.

non-volatile storage. Typically refers to non-volatile memory on a processor rather than a non-volatile storage device. On a storage facility, non-volatile storage is used to store active write data to avoid data loss in the event of a power loss.

NQ. node qualifier.

NVS. non-volatile storage.

O

OEMI. original equipment manufacturer's information. A reference to an IBM standard for a computer peripheral interface. More specifically, a reference to *IBM S/360 and S/370 Channel to Control Unit Original Equipment Manufacture's Information*. The interface utilizes ESA/390 logical protocols over an I/O interface that configures attached units in a multi-drop bus topology.

P

path group. The ESA/390 term for a set of channel paths that are defined to a control unit as being associated with a single LPAR and as being in the grouped state (e.g. online to the host).

path-group identifier. The ESA/390 term for the

identifier that uniquely identifies a given LPAR. The path-group identifier is used in communication between the LPAR program and a device to associate the path-group identifier with one or more channel paths, thereby defining these paths to the control unit as being associated with the same LPAR.

PCI.   program-controlled interrupt, peripheral-component interconnect.

Peer to Peer Remote Copy.   A facility of a storage server that maintains a consistent copy of a logical volume on the same storage facility or on another storage facility that is accessible over an ESCON I/O interface. All modifications of the primary logical volume by any attached host are also performed in order by the storage facility on the secondary logical volume. Options exist to request the modifications to be performed synchronously (i.e. before the completion of the modification of the primary copy is are completed).

| **PLOGI**.   port login

PPRC.   Peer to Peer Remote Copy.

predictable write.   A write operation that can cached without knowledge of the existing formatting on the medium. All writes on FBA DASD devices are predictable. On CKD DASD devices, a write is predictable if it does a format write for the first data record on the track.

program.   A generic term for the software that controls the operation of a host computer. Typically, the program is an operating system that allows sharing of the host resources between multiple tasks.

promote.   The action of adding a logical data unit to cache memory.

R

random access.   A mode of accessing data on a medium in a manner that requires the storage device to access non-consecutive storage locations on the medium.

reserved allegiance.   The ESA/390 term for a relationship that is created in a control unit between a device and a channel path, or path group, when a Sense Reserve command is completed by the device. The allegiance causes the control unit to guarantee access (i.e. busy status is not presented) to the device over the set of channel path associated with the allegiance, for one or more channel programs, until the allegiance is terminated.

R0.   track-descriptor record.

S

SBCON.   Single-Byte Command Code Sets Connection Architecture. An ANSI standard draft for the ESCON I/O interface.

SCSI.   Small Computer System Interface. An ANSI standard for a logical interface to computer peripherals and for a computer peripheral interface. The interface utilizes a SCSI logical protocol over an I/O interface that configures attached targets and initiators in a multi-drop bus topology.

Sequential Stage Group.   The number of tracks that will be prestaged in a sequential operation.

service processor.   A dedicated processing unit which is used to service a storage facility.

sequential access.   A mode of accessing data on a medium in a manner that requires the storage device to access consecutive storage locations on the medium.

SIM.   service-information message.

SSA.   Serial Storage Architecture. An IBM standard for a computer peripheral interface. The interface utilizes a SCSI logical protocol over a serial interface that configures attached targets and initiators in a ring topology.

SSID.   subsystem identifier.

stacked status.   An ESA/390 term for status that the control unit is holding for the channel because the channel responded with the stack-status control the last time the control unit attempted to present the status.

STI.   self-timed interface. An interface that consists of one or more conductors that transmit information serially between two interconnected units without requiring any clock signals to recover the data. Clock recovery is performed independently on each serial data stream and information in the data stream is used to determine character boundaries and inter-conductor synchronization.

storage device.   A physical unit which provides a mechanism to store data on a given medium such that it can be subsequently retrieved.

storage facility.   A physical unit which consists of a storage server integrated with one or more storage devices to provide storage capability to a host computer.

storage server.   A physical unit which provides an interface between one or more storage devices and a host computer by providing the function of one or more logical subsystems. The storage server may provide

functions that are not provided by the storage device. The storage server is composed of one or more clusters.

stripe.   The set of tracks in a RAID Rank that have the same parity track. Full stripe writes reduce the RAID 5 write penalty.

subchannel.   A logical facilities of a channel subsystem associated with the management of a single device.

subsystem identifier.   A number that uniquely identifies a logical subsystem within a computer installation.

synchronous write.   A write operation whose completion is indicated after the data has been stored on a storage device.

T

target.   The SCSI term for a storage server.

track.   A unit of storage on a CKD device consisting that can be formatted to contain a number of data records. Also see *home address, track-descriptor record,* and *data record.*

track-descriptor record.   A special record on a track following the home address that is used by the control program to maintain certain information about the track. The record has a count field with a key length of zero, a data length of 8, and a record number of 0. This record is sometimes referred to as R0.

U

unit address.   The ESA/390 term for the address associated with a device on a given control unit. On ESCON interfaces, the unit address is the same as the device address. On OEMI interfaces, the unit address specifies a control unit/device pair on the interface.

utility device.   The ESA/390 term for the device used with the Extended Remote Copy facility to access information describing the modifications performed on the primary copy.

V

volume.   The ESA/390 term for the set of information recorded on a single unit of recording medium or, indirectly, refers to the unit of recording medium itself. On a non-removable medium storage device, the terms may also refer, indirectly to the storage device associated with the volume. When multiple volumes are stored on a single storage medium transparently to the program, the volumes may be referred to as logical volumes.

VPD.   vital product data.

X

XRC.   Extended Remote Copy.

Index

T

U

V

W

Communicating Your Comments to IBM

IBM Enterprise Storage Server™
SCSI Command Reference
2105 Models E10, E20, F10, and F20

Publication No. SC26-7297-01

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:
    - United States: 1-800-426-6209
    - Other countries: (+1)+408+256-7896
- If you prefer to send comments electronically, use this network ID:
    - IBMLink from Canada: STARPUBS at TORIBM
    - IBM Mail Exchange: USIB3VVD at IBMMAIL
    - Internet: starpubs@us.ibm.com

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

Readers' Comments — We'd Like to Hear from You

IBM Enterprise Storage Server™
SCSI Command Reference
2105 Models E10, E20, F10, and F20

Publication No. SC26-7297-01

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

_____

Name

_____

Company or Organization

_____

Phone No.

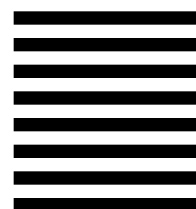_____

Address

Fold and Tape   **Please do not staple**   Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL   PERMIT NO. 40   ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
RCF Processing Department
G26/050
5600 Cottle Road
SAN JOSE, CA 95193-0001

Fold and Tape   **Please do not staple**   Fold and Tape

IBM®